(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 794 645 A2

(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
10.09.1997 Bulletin 1997/37

(51) Int. Cl.^{6}: H04L 29/06

(21) Application number: 97300244.7

(22) Date of filing: 16.01.1997

(84) Designated Contracting States:
DE FR GB

(30) Priority: 16.01.1996 US 586259

(71) Applicant: General DataComm, Inc.
Middlebury Connecticut 06762 (US)

(72) Inventor: Semaan, Ghassan
Montreal, Quebec H3H 2N3 (CA)

(74) Representative: Gibson, Stewart Harry
URQUHART-DYKES & LORD,
Three Trinity Court,
21-27 Newport Road
Cardiff CF2 1AA (GB)

(54) **Multimedia multipoint telecommunications reservation systems**

(57) Reservation controllers and reservation systems for reservation of access to multimedia multipoint telecommunications servers (MCUs) are provided. The reservation controller establishes a reservation domain having a reservation request channel on which reservation applications may be taken. An ongoing reservation conference is associated with the reservation domain. The reservation domain is preferably established within the ITU-T T.120 standard series requirements, with multipoint connection under T.122/T.125, and interface to profiles such as ISDN, TCP/IP, X.25, ATM, Ethernet, etc., under T.123. A user makes reservations for MCU resources by knowing the address of the reservation domain, attaching himself to the reservation domain, joining the reservation conference via establishing one or more transport connections, and sending the reservation request onto the reservation request channel. All reservation requests forwarded onto the reservation request channel are received and acted upon by the reservation controller. Reservation systems utilizing a plurality of reservation controllers are also disclosed, where the reservation controllers are part of the same domain, are arranged in multiple domains on a single level, or are arranged in hierarchical domains. Where more than one reservation domain is established, same-level or hierarchical level bridges are established.

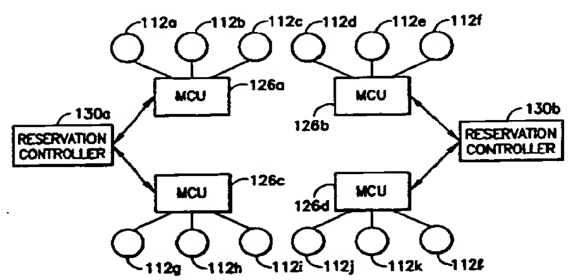


FIG.2

EP 0 794 645 A2

Description

BACKGROUND

1. Field of the Invention

The present invention relates broadly to multipoint multimedia telecommunications systems. More particularly, the present invention relates to reservation systems for the use of multimedia multipoint servers which are used in the establishment and conduct of multipoint multimedia conferences.

2. State of the Art

With the increase of throughput (data rate) available in the telecommunications industry, and in association with the improvement of compression and decompression algorithms, the number of telecommunication applications available to individuals and businesses has increased dramatically. One of these applications is called "multimedia communications" which permits video, audio, and in some cases other data to be transported from one party to another or others. Multimedia communications can be utilized for a number of applications, and in different configurations. One configuration of recent interest has been multimedia conferencing, where several parties can communicate in a conference style.

In multimedia conferencing, the audio and video data is handled such that each party can see and hear one, several, or all of the other parties. In fact, various telecommunications recommendations and standards are presently being adopted by the ITU-T, ISO, and Bellcore which govern the protocols of multimedia conferencing (see, e.g., ITU-T T.120). In the multimedia conferencing systems of the art (as represented by prior art Figure 1), the audio, video, and other data streams generated by a user's system 12a are multiplexed together directly in the encoder section of a multimedia encoder/decoder (codec) 14 located at the source/terminal 16, and transported together through the transport network 20 (now proposed in ATM format) to a similar "peer" codec at a remote location. The peer codec is either another codec 14 at the remote user site for a point-to-point conference, and/or a codec/switch 24 at a multimedia bridge 26 (also called a multipoint control unit or MCU) for a multipoint conference. The multipoint control unit 26, which typically includes a codec/switch 24 and a controller 28, provides conference control (e.g., it determines the signal to be sent to each participant), audio mixing (bridging) and multicasting, audio level detection for conference control, video switching, video mixing (e.g., a quad split, or "continuous presence device" which combines multiple images for display together) when available and/or desirable, and video multicasting. The audio and video data exiting the MCU is multiplexed, and continues through the transport network 20 to the desired multimedia source terminals 12b, 12c.

It will be appreciated by those skilled in the art that the MCU is a technically complex and expensive piece of equipment/system, and that use of the MCU is carefully controlled and billed to the user. In addition, it will be appreciated that because of its expense and limited availability, access to the MCU is treated as a rare resource. Thus, in order to guarantee to a given user the necessary resources for a desired conference at a given time, the user must reserve access to the MCU in advance of use. Reservations are made by a "reservation request" which typically involves a telephone call to an operator at the company with the MCU. In response to a reservation request which will often include parameters such as a starting time, a duration, and resources necessary (e.g., bandwidth, mixing and switching, etc.), the operator will typically access a "reservation controller", which is typically a programmed computer, in order to determine whether the required resources of the MCU will be available at the desired time. If the resources are available, the operator will enter information into the reservation controller program related to the incoming request for desired connections and services for the given time, accept the reservation, and inform the applicant/user of codes (e.g., a reservation number) required for the conference. When the time is reached for the conference, the reservation controller will inform the MCU of the beginning of a new conference and the precise resources reserved for that conference. When the users wish to join the conference and access the MCU, the users call the operator, provide the reservation number, and are added to the conference, with the necessary resources having been already reserved and available.

To date, the use of MCU's has been very limited. There are several reasons for this limited use. First, all multimedia services are extremely new, and most telecommunications customers have not yet invested in multimedia service equipment. It is expected, however, that the next five years will see an explosion of growth in the area of multimedia telecommunications. Second, the companies which provides multimedia equipment often utilize proprietary or alternative standardized schemes (e.g., MPEG and JPEG) which are not necessary compatible. Thus, it is often impossible for owners of different types of multimedia equipment to communicate with each other. Again, solutions to this problem are being proposed, including transcoders which are intended to make MPEG and JPEG equipment compatible. Third, and with respect to multipoint multimedia services such as conferencing, each manufacturer of an MCU uses its own proprietary reservation controller mechanism for reserving access to the MCU. Thus, unless each user in the conference has access to the same MCU, or to MCUs which are under control of the same reservation controller, it may be impossible for users to conference as desired because reservations requiring access to different MCUs may be impossible because of their control by different controllers.

SUMMARY OF THE INVENTION

It is therefore an object of the invention to provide a reservation controller for multimedia multipoint servers.

It is another object of the invention to provide a standardized reservation system which will permit any multimedia user to reserve the resources of one or more MCUs for a multimedia multipoint conference.

It is a further object of the invention to provide a reservation system for multimedia multipoint servers which utilizes the environment of the T.120 protocol series in receiving a reservation request.

It is an additional object of the invention to provide a multimedia multipoint server reservation system which has an established reservation domain which includes one or more reservation controllers and to which a user can attach in order to place a reservation application.

Another object of the invention is to provide an automatic multimedia multipoint reservation system.

A further object of the invention is to provide a multimedia multipoint server reservation system with multiple reservation controllers arranged in a hierarchical multiple domain structure.

An additional object of the invention is to provide a reservation domain having a reservation request channel utilized by multiple reservation controllers of a multimedia multipoint server reservation system.

Yet another object of the invention is to provide a server reservation system for multimedia multipoint communications with multiple reservation domains having continuous and/or dynamic connections.

In accord with the objects of the invention, and according to a first primary aspect of the invention, a reservation controller for use with one or more multimedia multipoint servers (MCUs) is provided where the reservation controller establishes a reservation domain with a reservation request channel on which reservation applications may be taken. In the preferred embodiment, the reservation domain is established within the T.120 standard series requirements (with the term "domain" being defined in ITU-T T.122, and loosely defined as including all members who are connected to a conference), with multipoint connection under T.122/T.125 (Multipoint Communications services), and interface to profiles such as ISDN, TCP/IP, PSDN (X.25), ATM, IEEE 802.3 (Ethernet) etc., under T.123 (Transport Protocol Stack Profiles). If desired, conference control under T.124 (Generic Conference Control) may also be provided, although other conference control mechanisms can be utilized. In accord with the invention, and based on the T.120 series of standards, a reservation "domain" with a reservation request channel is established within T.122/T.125 by the reservation controller in order to collect point-to-point transport connections and combine them to form a multipoint connection. An ongoing reservation conference is associated with the reservation domain. If the reservation controller establishes a reservation conference under T.124, the reservation domain is set up as part of the reservation conference.

When users wish to make reservations for resources of the multimedia multipoint server, they must know the address of the reservation domain, and must attach themselves to the reservation domain (the term "attach" also being defined in T.122) and join the reservation conference. Typically, this will be accomplished by establishing one or more transport connections (as defined by ITU-T standards X.214 and X.224 which are hereby incorporated by reference in their entirety herein) between the user and the reservation controller. The transport connection could be established either utilizing X.214 and X.224, or via the use of other protocols such as X.25, TCP/IP, ATM, etc., many of which are defined in ITU-T T.123. At the lower layers, any type of protocol can be used such as Ethernet, ISDN, PPP, etc.) Once attached to the reservation domain, the user can make a reservation request by sending the reservation request onto the reservation request channel. All reservation requests forwarded onto the reservation request channel are received and acted upon by the reservation controller. The reservation controller is coupled to one or more MCUs and preferably stores the schedules of the MCUs to which it is coupled so that it can properly act on the reservation request.

In the situation where multiple reservation controllers are provided, as discussed below, the user calls the address domain of the "local" reservation controller in order to attach to the reservation domain and join the reservation conference. Once joined to the reservation conference, the user can then send the reservation request onto the reservation request channel of the reservation domain. By utilizing the reservation request channel, when a reservation request which affects MCUs of different reservation controllers is placed on the channel, each affected reservation controller can determine whether the MCU or MCUs for which it is responsible is/are available as requested. Preferably, the resulting determination of each controller is provided to the reservation controller of the MCU most local to the user for processing and forwarding to the user. Alternatively, each reservation controller can send information regarding the availability of its MCUs to the user.

In accord with a second primary aspect of the invention, a reservation system is provided and includes a first plurality of reservation controllers which are parties to a first reservation domain; i.e., the reservation controllers are all coupled to a first reservation request channel. The reservation system may be further expanded by including a second plurality of reservation controllers which are parties to a second reservation domain, where one or more reservation controllers can act as a bridge between the different domains. The bridge or connection between the domains can be static (continuous), or dynamic (connected intermittently on an as-needed basis). If desired, the reservation domains may be hierarchical; i.e., providing different levels. Thus, the reservation domains may parallel the

local/regional/national/international telephone system, with first level domains relating to local MCUs, second level domains relating to regions (e.g., within a single area code); third level domains relating to countries; and fourth level domains relating to international conferences. The connections between the hierarchical domains can likewise be static or dynamic. It should also be appreciated that regardless of how the reservation domains of the reservation system are connected, and whether or not they are hierarchical, the reservation domains which are established are preferably established under the T.120 standard series requirements.

According to additional preferred aspects of the invention, each user receives reservation information (e.g., answers to reservation requests) on the user's private channel. Also, besides utilizing a reservation requests channel for transmission of reservation applications in a multi-reservation controller system, additional channels can be provided. For example, one or more channels dedicated for the transmission of management information between one or multiple reservation controllers and their associated MCUs may be provided. Likewise, one or more channels used solely for the transmission of reservation data amongst the different controllers may be provided.

Additional objects and advantages of the invention will become apparent to those skilled in the art upon reference to the detailed description taken in conjunction with the provided figures.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a representation of a multimedia conferencing system of the prior art, including a telecommunications network and an MCU.

Figure 2 is a high level diagram showing a system with a plurality of users, a plurality of MCUs, and a plurality of reservation controllers.

Figure 2a is a high level diagram showing the system of Figure 2 with the reservation domain and reservation request channel of the invention.

Figure 3 is a diagram of the ITU-T T.120 series Infrastructure Recommendation modified in accord with the invention to show a reservation application.

Figure 4 is a diagram of a ITU-T T.120 generic application model on which the reservation application of the invention is based.

Figure 4 is a diagram of the server part of the reservation application of the invention.

Figure 5 is a high level diagram of a single level reservation system according to the invention.

Figure 6 is a high level diagram of a hierarchical reservation system with multiple levels according to the invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Turning to Figure 2, a hypothetical telecommunications system is shown and includes a plurality of users 112a-112l (typically coupled to the network 20 - see Fig. 1), a plurality of MCUs 126a-126d (typically located in the network), and a plurality of reservation controllers 130a, 130b (typically coupled to and/or located at the MCUs). Each reservation controller 130a, 130b, is shown coupled to two MCUs, while each MCU is shown servicing three users. It will be appreciated by those skilled in the art that depending upon its configuration and the needs of the users, each MCU can service many more than three users; and depending upon similar parameters, each reservation controller can service more than two MCUs. However, for purposes of simplicity of understanding, two reservation controllers, four MCUs, and twelve users are shown. With the provided arrangement, it will be appreciated that if users 112c, 112e, 112f, 112g, 112h, and 112j should wish to participate in a multimedia conference, the services of the four different MCUs 126a-126d will be required. Thus, the two reservation controllers 130a, 130b must be contacted to reserve appropriate access and processing of the MCUs. However, with the systems that presently exist in the art, if the MCUs and reservation controllers are owned and operated by different companies, it may be impossible to arrange such a multimedia conference. In addition, with the provided arrangement it is not evident to which reservation controller the user should forward a reservation request, and how the reservation controllers will share the information contained in the request among themselves.

As seen in Fig. 2a, and in accord with the invention, one of the reservation controllers 130a, 130b of the hypothetical telecommunications system of Fig. 2 initiates and establishes a "reservation domain" which is provided with (typically upon request) a "reservation request channel" 135; and the other of reservation controllers attaches itself to the established reservation domain and joins the reservation request channel 135. The reservation domain is associated with a "reservation conference" (which if desired, may be pursuant to ITU-T T.124) and attachment to the reservation domain is accomplished by joining the conference. As is defined by ITU-T T.122 (MCS), any node (users, reservation controllers, MCUs, etc.) which attaches itself to a domain will be assigned a private channel (also called a "single member channel"). The private channel is normally used as a user identifier which provides a user identification and serves as an address for point-to-point communication within the multipoint domain. However, within T.122, another type of channel called a multicast channel can be defined to which any number of nodes can be joined. The reservation channel 135 is such a multicast channel.

It should be appreciated by those skilled in the art that any node which is attached to the domain can send data on any channel in the domain (the ability to send data being shown by dotted lines in Fig. 2a). However, only nodes which have joined a particular channel will receive the data sent on that particular channel (the

ability to receive data being shown by solid lines in Fig. 2a). Thus, any node wishing to send private data to any other node will do so by sending this data on the private channel of the destination node, with representative private channels 147c and 147i being shown in Fig. 2a for users 112c and 112i.

A user who wishes to make a reservation will join the reservation conference and must attach himself to the reservation domain, but will not join the reservation request channel. Once attached to the reservation domain, the user can attempt to place a reservation by sending a reservation request onto the reservation request channel 135. The reservation request will typically include a plurality of multimedia conference parameters discussed in more detail below such as the starting time, the duration, the addresses of the users involved, and the resources necessary for the conference. In addition, the user will specify his own private channel address for reservation confirmation. Since the reservation controllers 130a, 130b are party to the reservation domain and have joined the reservation request channel, the parameters placed on the reservation request channel 135 are available to (i.e., are received by) the reservation controllers 130a and 130b.

Where the set of users who will be party to the multimedia conference all are serviceable by a single MCU (such as users 112d, 112e, and 112f), then the reservation controller (e.g., 130b) for that single MCU (e.g., MCU 126b) will make a determination as to whether the necessary MCU 126b resources will be available for the requested conference for the time requested. If so, the reservation controller 130b will confirm the reservation with the conference-initiating user via the private channel of the user. Where the set of users who will be party to the multimedia conference are not all serviceable by a single MCU (such as users 112a, 112b, and 112g), but a single reservation controller (e.g., controller 130a) is involved, again, the reservation controller can determine whether resources are available and can confirm the reservation with the conference-initiating user via the private channel of the user. However, where the set of users who will be party to the multimedia conference are serviced by multiple MCUs which are serviced by multiple reservation controllers, (such as users 112c, 112e, and 112f), then the reservation controller(s) (e.g., 130a, 130b) for the MCUs involved (e.g., MCUs 126a, 126b) will make determinations as to whether the necessary resources of the MCU 126a, 126b under their control will be available for the requested conference for the time requested.

Determination as to the availability of MCU resources which are governed by multiple reservation controllers can be accomplished in several ways. In one preferred embodiment where all of the reservation controllers are joined to the reservation request channel, and they all receive every reservation request which is sent to the channel, each controller is programmed to determine the part or parts of the resources requested by the user which are managed by itself. Thus, each reservation controller can proceed to process the part of the request for which it is responsible (i.e., a sub-request). After processing their sub-requests, those reservation controllers which are not the master reservation controller (i.e., the master being the most local reservation controller to the user) for the request send responses to the reservation controller which is the master for the request. After reviewing the responses, the master informs the user of the acceptance or refusal of the request on the private channel of the user, and informs the other reservation controllers of the result of the request. When a reservation is confirmed, the reservation controllers involved update their MCU resource files.

In a first alternative approach, rather than having each reservation controller respond to the master, each reservation controller can directly inform the user of its response. The reservation process running at the user's terminal would then be used to gather all of the responses, so that the user could determine whether the reservation (or a portion thereof) could be accepted. If the resources proved to be suitable, the user would place a confirmation on the reservation request channel, and the reservation controllers would update their MCU resource files accordingly.

As a second alternative approach, the reservation controller responsible to answer the request (i.e., the master for the request), can communicate with the other reservation controller(s) and ask for the status of the MCU resources for which they are responsible. Upon gathering the information, the master can decide whether to accept the reservation or not, inform the user of the decision, and inform the other controllers so that they can update their MCU resource files if necessary.

It should be appreciated that besides requesting a new reservation, the user will typically have the capability of making other requests. For example, the user should be able to view, cancel, and modify a previously accepted reservation. Furthermore, the user preferably should be able to place queries regarding the availability of resources.

According to the preferred embodiment of the invention, the entire reservation system, including the reservation domain and the reservation request channel are generated in accordance with and comply with the ITU-T T.120 series of standards (the latest versions of which are all hereby incorporated by reference herein in their entirety). T.120 and T.121 define the relationship between a T.12x application and the remaining T.12x recommendations or standards, with 7.120 defining the environment, and T.121 defining the Generic Application Model to be used with the applications which use T.122/T.125 (Multipoint Communications Service or MCS) and T.124 (Generic Conference Control or GCC). Therefore, to be compatible, a "reservation application" under T.120 must be built with respect to at least the T.120, T.121, and T.122/T.125 recommendations. Because audio and video control are not particularly pertinent to the reservation application, use of T.124 is

optional, although minimal control is required for the multipoint application.

As will be appreciated from Fig. 3, and in accord with the T.120 infrastructure recommendations, network specific transport protocols are set forth in T.123 which permit interface with profiles such as Ethernet (IEEE 802.3), TCP/IP (the Internet), X.214/X.224, ATM, etc. In addition, it should be appreciated that "agents" can be provided which will translate or otherwise transform commands and data of other profiles which are not part of T.123 so that interface with T.122/T.125 is possible.

Sitting atop the T.123 interface is the MCS T.122/T.125 standard which governs multipoint communications. Various multipoint applications can be implemented utilizing T.122/T.125. It should be appreciated that T.122/T.125 is particularly desirable and suitable for the reservation application of the invention (shown in dotted lines in the applications section of Fig. 3 to indicate that the reservation application is not yet a standard application of the ITU-T), as the reservation application of the invention utilizes multipoint communications. In addition, as shown in Fig. 3, the GCC T.124 standard can be provided for conference control, although the use of conference control under T.124 is not necessary for practicing the invention. If GCC is provided, however, a node controller is required for gathering requests from the applications and sending them to the GCC. It is noted that Fig. 3 shows that MCS and GCC can support applications using standard application protocols T.12x (such as T.126 Still Image and T.127 File Transfer), as well as applications using non-standard protocols and applications using standard and non-standard protocols.

As will be appreciated by those skilled in the art, and as suggested by Figure 4, the T.121 standard requires that applications include two parts: a User Application (UA), and one or more Application Protocol Entities (APEs). An APE is further divided into two elements: an Application Resource Manager (ARM), and an Application Service Element (ASE).

The User Application has no direct affect on interworking and thus may be product and platform specific. The User Application relies on the services of one or more APEs to communicate with peer applications at other nodes, and does not directly communicate with the MCS (T.122/T.125) and the GCC (T.124) shown in Fig. 4. According to a preferred embodiment of the invention the reservation UA is programmed to include the following functionalities: reservation type, address list, repeating and periodic conferences, help menu, and input validity checking. The reservation type functionality is a program which permits the user to indicate whether a new reservation is being requested, or whether a review, change or cancellation of a prior reservation is being requested. The address list functionality is a program which permits and/or requires the user to enter a list of addresses, write cascaded lists, edit address lists on-line, name a list, edit a previously made list, label each address in the list, and specify which entry in the list will have chairperson/broadcaster control.

The UA can also specify parameters which are to be provided by the customer. Among the preferred customer-provided parameters are: account billing, number of conferees, conference timing, conference setup mode, conference mode, conference quality, physical channel selection, and video receiving mode. The account billing permits the user to enter the calling number or card number which is to be the billed account. The number of conferees who will participate in the conference should be fixed at reservation time due to resource allocation management, although the user should be able to modify this number before or during the conference, provided resources are free at the MCUs involved. The date, time and duration of the conference should also be fixed at reservation time due to resource management, with the duration also being alterable before or during the conference subject to resource availability. In conjunction with duration parameter, the user may be able to specify a conference termination mode.

The conference setup mode is the mode used by the conferees to join a conference and may specify a parameter (whether the conference is listed or not) and three lists. When the parameter indicates that the conference is not listed, only the conferees specified in the three lists can joint the conference. Otherwise, any conferee with the proper password can join the conference. The three lists preferably include a list of users that will be automatically called when the conference is created, a list of customers who have permission to joint the conference after it is created provided that they have the proper password, and a list of customers who have permission once they are conferees to invite other customers to joint the conference. Each list can contain data or be empty.

The conference mode should include the four known basic modes of voice-activated switching, chairperson control, broadcast monologue, and broadcast dialogue. In addition, if desired, other modes such as conferee's choice, automatic control, and subconferencing can be supported. In conferee's choice, each conferee can decide whom (s)he wants to see and whom s(he) wants to hear. In automatic control, the MCU will decide what must be seen by whom as well as what must be heard. If subconferencing is enabled, two or more participants to a conference will be able to initiate a "private" conference while they are still members of the initial conference.

The conference quality parameter provided by the user specifies the video and audio qualities (bandwidths) desired, as well as the bandwidth desired for other data. Video quality may be constrained by the standard (e.g., JPEG or MPEG) being used, as well as local resources of the conferees, network capacity, etc. Likewise, audio quality may be constrained by the standard being used, and the application (audio vs. voice). The physical channel selection parameter pro-

vided by the user allows the user to specify and identify which video terminals will use circuit switching to connect to the MCU and which video terminals will use private lines to connect the MCU. Finally, the video receiving mode permits the user to select whether conferees will receive a single video feed (e.g., of the current speaker) from another conferee, a merged video feed (e.g., a quad split or "continuous presence"), or multiple video feeds which permits the conferee to individually configure and position the incoming images.

Turning now to the Application Protocol Entities (APE) side of the T.121 standard requirement, the Application Resource Manager (ARM) of the APE provides generic functionality common to all standardized application protocols, while the Application Service Elements (ASEs) provide functionality specific to their respective application protocols. As set forth by T.121, an APE is characterized by the following attributes: a single MCS service access point (SAP); a single GCC SAP, a single application user ID, a single ARM, a single ASE, and a node controller SAP.

The ARM is responsible for managing GCC and MCS resources on behalf of the ASEs within the APE. The ARM should provide the following services: responding to indications from the GCC (e.g., permission to enroll, invoke); enrolling ASEs with the GCC; obtaining handles from the GCC; attaching to an MCS domain to obtain a single Application User ID for all ASEs within the APE; joining static channels; identifying and joining multicast channels using the GCC Registry and MCS; convening private channels and admitting peer ASEs to such channels; joining any private channels to which an ASE has been admitted; identifying and obtaining tokens from the GCC Registry; deleting entries from the registry associated with any multicast channel it may have convened; invoking peer APEs at other nodes; and processing Application Roster reports to determine the negotiated Application Capability list and identity of peer nodes.

The ASE provides application protocol specific functionality to the user application with resources obtained by the ARM. Its operation is independent of the type and identity of tokens and channels passed to it. The User Application should specify the type of resources to use, but not the identity of those resources. The ASE obtains the identity of resources to use from its ARM.

The ASE provides the following services: sending and receiving application protocol-specific protocol data units (PDUs); grabbing and releasing tokens and determining token status using MCS; responding to GCC Conductor Assign and Release indications; issuing GCC-Conductor-Permission-Ask requests through the Node Controller; and responding to GCC-Conductor-Permission-Grant indications.

According to the preferred embodiment of the invention, the reservation application has a protocol which is divided into three parts: the user (conferee) part which is the part which runs on the user's terminal and allows the user to make a reservation; the MCU part which runs at the MCU; and the server part which runs at the reservation controller and processes all the reservation requests. These three separate parts represent a logical division of the protocol and not necessarily a physical one, as the physical MCU can have both the MCU and the server part of the protocol. In other words, the MCU and the reservation controller may be integrated physically.

The user part of the reservation application is the part that runs on the user's terminal. It allows the user to exchange data with the reservation server in order to make reservations. The tasks of this part are: to establish and maintain a MCS (T.122/T.125) connection via T.123 with a reservation controller; to accept, code, and send reservation requests and parameters of the user to the reservation controller via the reservation request channel; and to receive results from the reservation controller via the user's private channel and display them to the user.

The MCU part of the reservation application runs on the MCU and allows the MCU to communicate with the reservation controller to inform it of the status of its resources. The tasks of this part are to establish and maintain a connection with the reservation controller, and to receive and answer requests from the reservation controller regarding the status of the resources of the MCU and the starting of conferences.

The server part of the reservation application is the part that runs an the reservation controller which is preferably embodied as a suitably programmed SUN SPARC STATION 5 having suitable memory (although other processors, computers, or work stations could be utilized). The server part of the reservation application which is shown schematically in Fig. 4a allows the controller to exchange data with the users and the MCUs in order to process the different requests of the users/conferees. The tasks of this server part are to: establish and maintain multiple simultaneous MCS connections with users, MCUs, and other reservation controllers (via use of the application resource manager 170, the MCS 172, and if desired, the GCC 174), thereby establishing a reservation domain and conference; monitor the status of all MCUs connected to the reservation controller, including notifying the MCUs of the start and the resources required for a conference (using the resource handler 182); receive, process, and accept or refuse the requests of the users such as new reservations, modifications of reservations, etc. based on any desired acceptance algorithm (using the reservation manager 184 and the acceptance algorithm 186); maintain a list of the reservations for which it is responsible (at the reservation handler 188 of Fig. 4a); and exchange data with other reservation controllers when a reservation request necessitates the use of more than one reservation controller. It will be appreciated by those skilled in the art, that the algorithm for accepting or rejecting new reservations, or modifications of the reservations can be proprietary to the supplier of the reservation controller.

Turning now to Fig. 5, a hypothetical system with ten reservation controllers 230a-230j, each of which is respectively coupled to an MCU 226a-226j and two users 212a1-212j2 is seen. The reservation controllers of the reservation system of Fig. 5 are structured on a single level, in that any reservation controller can connect itself to any other reservation controller and exchange data with it. It should be appreciated, that in accordance with the invention, the single level structure of the system of Fig. 5 can utilize a single reservation domain, or multiple reservation domains. In particular, if all of the reservation controllers 230a-230j are attached to the same reservation domain, then they will all be joined to the same reservation request channel. Thus, any reservation request which is placed on the reservation request channel will be reviewed by each reservation controller of the system for purposes of determining whether its resources will be required. On the other hand, where numerous reservation controllers are being utilized, it might be advantageous to break the reservation controllers into more than one domain. In this situation, at least one of the reservation controllers should act as bridge by being part of two or more reservation domains. Thus, if a user should wish to establish a conference with conferees who would be handled by the reservation controller of another domain, the bridge controller would pass the reservation request information onto the reservation request channel of the other reservation domain so that the appropriate reservation controller in the other domain could address the request. In Figure 5, a single level, two domain system is seen, where reservation controllers 230a-230e are part of a first domain, and reservation controllers 230e-230j are part of the second domain. Since reservation controller 230e is part of both domains, it acts as the bridge controller.

Turning to Figure 6, the same ten reservation controllers 230a-230j, MCUs 226a-226j, and users 212a1-212j2 discussed above with reference to Fig. 5 are seen, but they are arranged in a multiple level or hierarchical architecture. In a hierarchical system, multiple domains are established. While many different hierarchical systems can be provided, the system of Figure 6 suggests a preferred approach where each reservation controller (with each controller possibly corresponding to a plurality of local exchanges) establishes its own reservation domain with one or more local MCUs and the local users. A plurality of second level reservation domains (each corresponding, if desired, to an area code) are established by providing second level reservation controllers 330a-330e which share a reservation request channel with one or more of the first level reservation controllers. Thus, as seen in Fig. 6, reservation controllers 230a and 230b are conferenced with second level controller 330a in a first second level reservation domain; reservation controllers 230c and 230d are conferenced with second level controller 330b in a second second level reservation domain; reservation controllers 230e, 230f, and 230g are conferenced with second level controller 330c in a third second level reservation domain; reservation controller 230h is conferenced with second level controller 330d in a fourth second level reservation domain; and reservation controllers 230i and 230j are conferenced with second level controller 330e in a fifth second level reservation domain.

A third level reservation domain is shown in Fig. 6 by the provision of a third level controller 430 which conferences with all of the second level reservation controllers 330a-330e. The third level domain can correspond, if desired, to a country code. It should be appreciated that a fourth level (e.g., international) domain may also be established. In fact, by hierarchically dividing the domains in different manners, it will be appreciated that many more than four levels can be provided.

The advantage of the hierarchical architecture of Fig. 6 over the single level architecture of Fig. 5 is that there is no need for a bridging reservation controller to keep large amounts of information regarding the different systems which it is bridging. Rather, if a reservation controller notes a request on the reservation request channel of its domain which it cannot handle, it immediately passes that information to the reservation controller of the higher domain to which it is also party. Depending upon whether or not the information can be handled at that level, the reservation controller of the higher domain may then either pass the information to yet a higher domain, or back down to the appropriate lower domain. Thus, in the hierarchical architecture, the reservation controllers bridge different levels of domain, but need not keep detailed information regarding the higher domains.

It should be appreciated by those skilled in the art that the architectures of Figures 5 and 6 are not necessarily exclusive of each other. In other words, it is possible to use single level bridging reservation controllers as part of one or more levels of a hierarchical arrangement. It will also be appreciated that the architecture of the reservation application can be based on either a "continuous connection" or a "dynamic connection" type arrangement. In the continuous connection arrangement, all nodes that are concerned by reservations, as well as the nodes that might need one time to make a reservation must be connected continuously to the reservation domain. The advantage of the continuous connection arrangement is that MCS connection time is minimized. With the dynamic connection arrangement, the nodes establish different connections based on their instant needs, with connections being made each time a conferee has a reservation request, and connection being released once the processing of the request is terminated. With the dynamic connection arrangement, a connection is created (i.e., the domain is modified) for each request, and the connection is deleted (i.e., the domain is modified again) when the result of the request is known. The advantage of the dynamic connection arrangement is that the cost of connection is limited only to the time of actual usage. Again, it should be recog-

nized that the continuous connection and dynamic connection arrangements may be co-utilized. For example, the domains which include the users may be set up as dynamic connection reservation domains, as most users will not need continuous type connections and will find it expensive to keep a connection active when nobody is transmitting data. On the other hand, the domains which include only reservation controllers (and MCUs) may be set up as continuous connection reservation domains, as large amounts of data may be regularly sent, and connection may be required almost continuously. If desired, statistical measurements may be made of the use of each connection and therefrom a decision can be made as to whether the dynamic or the continuous connection is most desirable for that connection. In fact, if desired, some connections may be continuous during periods of peak usage, and dynamic at other times.

There have been described and illustrated herein multimedia multipoint telecommunications reservation systems. While particular embodiments of the invention have been described, it is not intended that the invention be limited thereto, as it is intended that the invention be as broad in scope as the art will allow and that the specification be read likewise. Thus, while the invention has been described with reference to the ITU-T T.120 family of standards, it will be appreciated that other platforms could be utilized provided that a reservation domain is established with a reservation request channel to which users can connect to make a reservation request. Likewise, while the invention was described with reference to particular arrangements with only one or two MCUs coupled to each reservation controller, and only two or three users coupled to an MCU, it will be appreciated that different numbers of MCUs could be used in conjunction with a reservation controller, and different numbers of users could be coupled to an MCU. In fact, pursuant to T.122/T.125, up to 65,535 connections can be made in a single domain, although it would not be deemed advisable to have so many users connected to a single MCU. Also, while the conference control was described with reference to T.124, it will be appreciated that since only minimal control is necessary for the reservation application (i.e., data only as opposed to audio/video/data), other control mechanisms which do not meet T.124 requirements could be utilized. It will therefore be appreciated by those skilled in the art that yet other modifications could be made to the provided invention without deviating from its spirit and scope as so claimed.

Claims

1. A reservation controller for controlling access to at least one telecommunications multimedia multipoint control unit (MCU), comprising:

   a) means complying substantially with ITU-T T.122 multipoint communications service (MCS) protocol for establishing with multipoint connection a reservation domain with a reservation request channel;

   b) conference control means associated with said reservation domain and establishing an ongoing reservation conference;

   c) first interface means complying substantially with at least a portion of ITU-T T.123 network specific transport protocol for interfacing with a data transport profile of a user who wishes to make a reservation;

   d) second interface means for coupling said reservation controller to the at least one MCU;

   e) reservation application protocol entity means including means for receiving a reservation request placed on said reservation request channel, and means for determining whether the at least one MCU coupled to said reservation controller has sufficient resources to meet said reservation request, wherein

   the user makes said reservation request by attaching to said reservation domain, joining said ongoing reservation conference, and placing said reservation request on said reservation request channel.

2. A reservation controller according to claim 1, wherein:

   said first interface means complies substantially with all of said ITU-T T.123 network specific transport protocol and provide an X.214/X.224 protocol interface, an IEEE 802.3 protocol interface, an ATM protocol interface, and a TCP/IP protocol interface.

3. A reservation controller according to claim 1, wherein:

   said conference control means complies substantially with at least a portion of ITU-T T.124 generic conference control (GCC) protocol, and controls said ongoing reservation conference.

4. A reservation controller according to claim 1, wherein:

   said reservation controller is coupled to a plurality of MCUs, and said reservation application protocol entity means includes means for storing resource schedules of said plurality of MCUs.

5. A reservation controller according to claim 1, further comprising:

   means for sending a response regarding said reservation request to the user via a private

channel of the user.

6. A reservation controller according to claim 1, further comprising:

means for collecting information from another reservation controller.

7. A reservation controller according to claim 6, wherein:

said information is a response by said the other reservation controller to a reservation request regarding an MCU coupled to the other reservation controller.

8. A reservation controller according to claim 6, wherein:

said information relates to a resource schedule of an MCU coupled to the other reservation controller.

9. A reservation system which controls access to a plurality of telecommunications multimedia multipoint control units (MCUs), comprising:

a) a first reservation controller having

means for establishing with multipoint connection a first reservation domain with a first reservation request channel,
conference control means associated with said first reservation domain and establishing a first ongoing reservation conference,
first interface means for interfacing with a data transport profile of a user wishing to make a reservation,
second interface means for coupling said first reservation controller to a first MCU of the plurality of MCUs the resources of which is controlled by said first reservation controller, and
first reservation application protocol entity means including first means for receiving a reservation request placed on said first reservation request channel, and means for determining whether the first MCU coupled to said first reservation controller has sufficient resources to meet its portion of said reservation request;

b) a second reservation controller having

means for joining said first ongoing reservation conference and for coupling to said first reservation request channel, wherein reservation requests of a user placed on said first reservation request channel are available to said second reservation controller when said second reservation controller is joined to said first ongoing reservation conference,
third interface means for coupling said second reservation controller to a second MCU of the plurality of MCUs the resources of which is controlled by said second reservation controller, and
second reservation application protocol entity means including second means for receiving a reservation request placed on said reservation request channel.

10. A reservation system according to claim 9, wherein:

said second reservation controller has means for determining whether the second MCU coupled to said second reservation controller has sufficient resources to meet its portion of said reservation request, and means for forwarding a resource determination to said first reservation controller.

11. A reservation system according to claim 9, wherein:

said second reservation controller includes means for sending information regarding resources of said second MCU to said first reservation controller.

12. A reservation system according to claim 10, wherein:

said second reservation controller has fourth interface means for interfacing with a data transport profile of another user wishing to make a reservation,

13. A reservation system according to claim 12, wherein:

said second reservation controller further includes means for establishing with multipoint connection a second reservation domain with a second reservation request channel, and second conference control means associated with said second reservation domain and establishing a second ongoing reservation conference.

14. A reservation system according to claim 9, wherein:

said second reservation controller further includes means for establishing with multipoint connection a second reservation domain with a second reservation request channel, and second conference control means associated with said second reservation domain and establishing a second ongoing reservation conference.

15. A reservation system according to claim 14, further comprising:

a third reservation controller coupled to at least one of said first and second reservation domains and having,

second means for joining at least one of said first and second ongoing reservation conferences and for coupling to at least one of said first and second reservation request channels, interface means for coupling said third reservation controller to a third MCU of the plurality of MCUs the resources of which is controlled by said third reservation controller, and
third reservation application protocol entity means including third means for receiving a reservation request placed on one of said first and second reservation request channels.

16. A reservation system according to claim 15, wherein:

said first reservation domain is at a first hierarchical level, and said second reservation domain is at a second hierarchical level.

17. A reservation system according to claim 16, wherein:

said third reservation controller is at one of said first and second hierarchical levels.

18. A reservation system according to claim 16, wherein:

said third reservation controller is at a third hierarchical level, and further includes means for establishing with multipoint connection a third reservation domain with a third reservation request channel, and third conference control means associated with said third reservation domain and establishing a third ongoing reservation conference.

19. A reservation system according to claim 16, wherein:

said means for joining said first ongoing reservation conference joins said second reservation controller to said first ongoing conference on an as-needed basis.

20. A reservation system according to claim 16, wherein:

said means for joining said first ongoing reservation conference joins said second reservation controller to said first ongoing conference on an ongoing static basis.

21. A reservation system according to claim 9, wherein:

said means for establishing with multipoint connection a first reservation domain, and said means for joining said ongoing first reservation conference and for coupling to said first reservation request channel both comply substantially with ITU-T T.122 multipoint communications service (MCS) protocol.

22. A reservation system according to claim 21, wherein:

said first interface means complies substantially with at least a portion of ITU-T T.123 network specific transport protocol for interfacing with a data transport profile of a user who wishes to make a reservation.

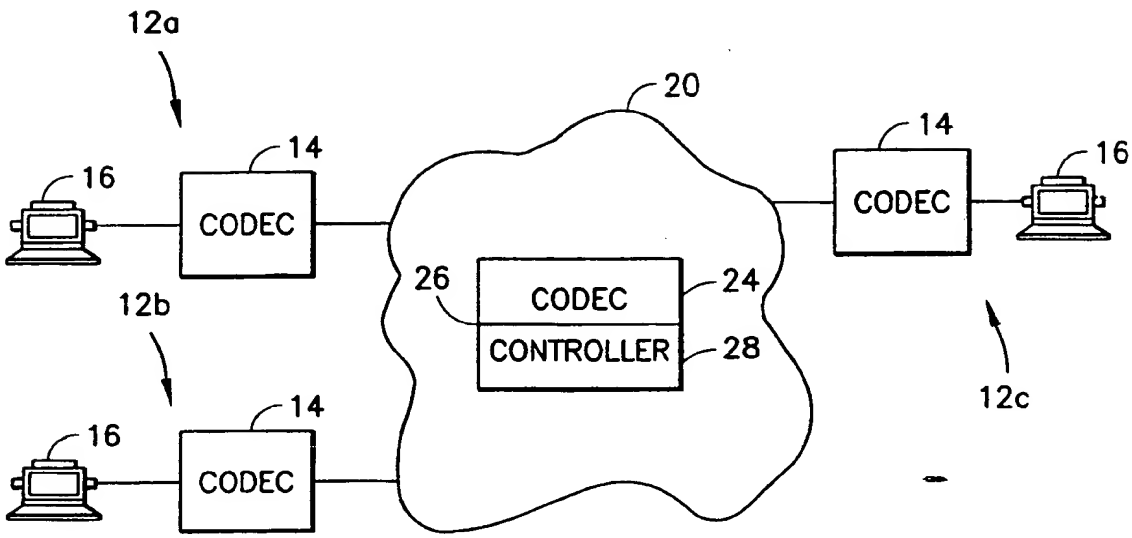


FIG.1
PRIOR ART

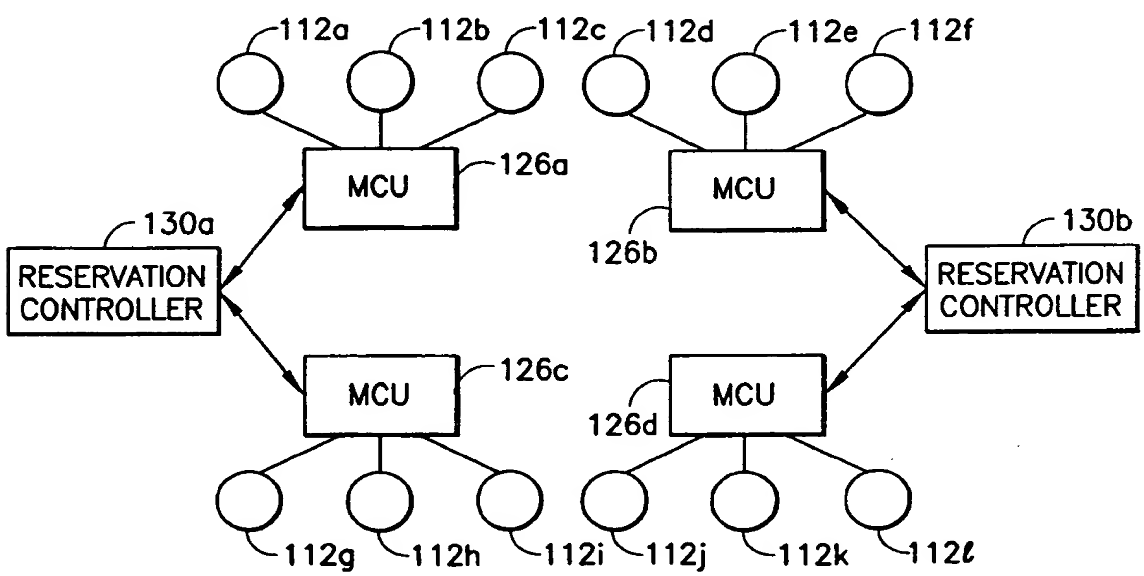


FIG.2

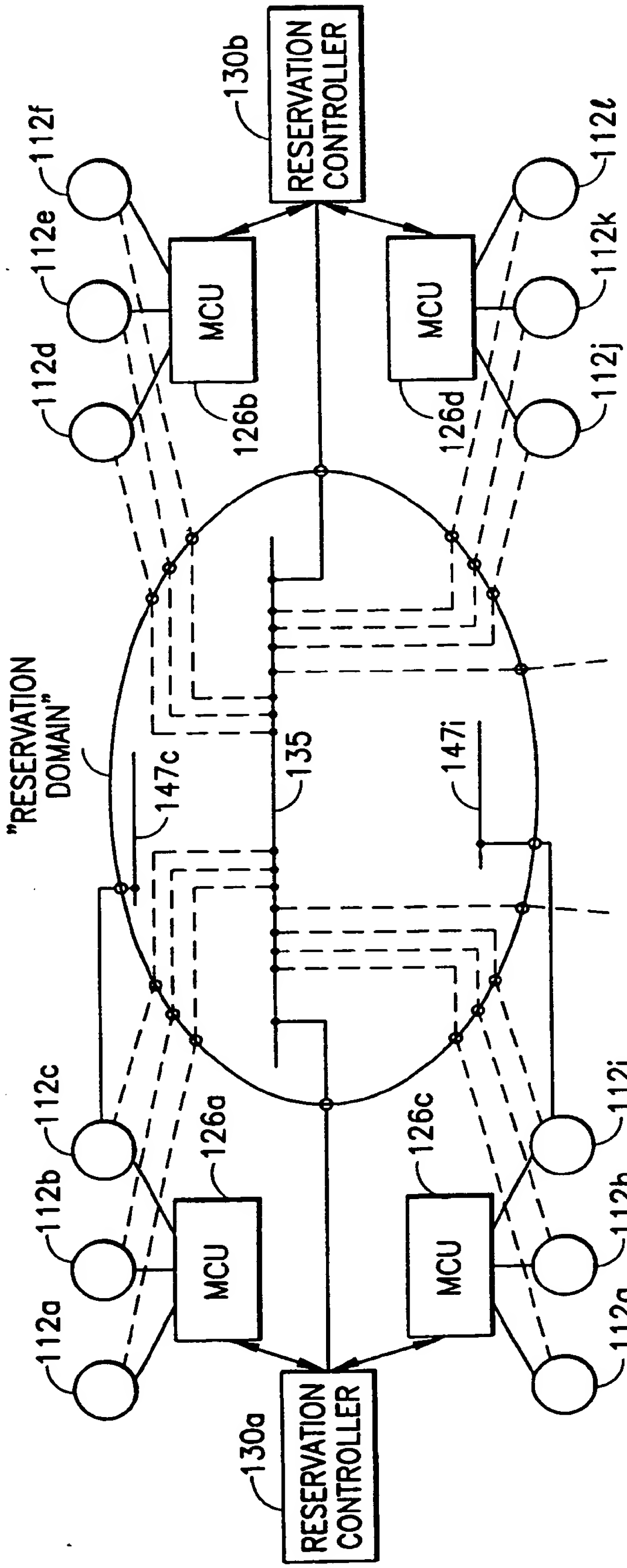

RESERVATION CONTROLLER
130b
112f
112e
112d
MCU
126b
112ℓ
112k
112j
MCU
126d
"RESERVATION DOMAIN"
147c
135
147i
112c
112b
112a
126a
MCU
126c
MCU
112i
112h
112g
130a
RESERVATION CONTROLLER

FIG.2a

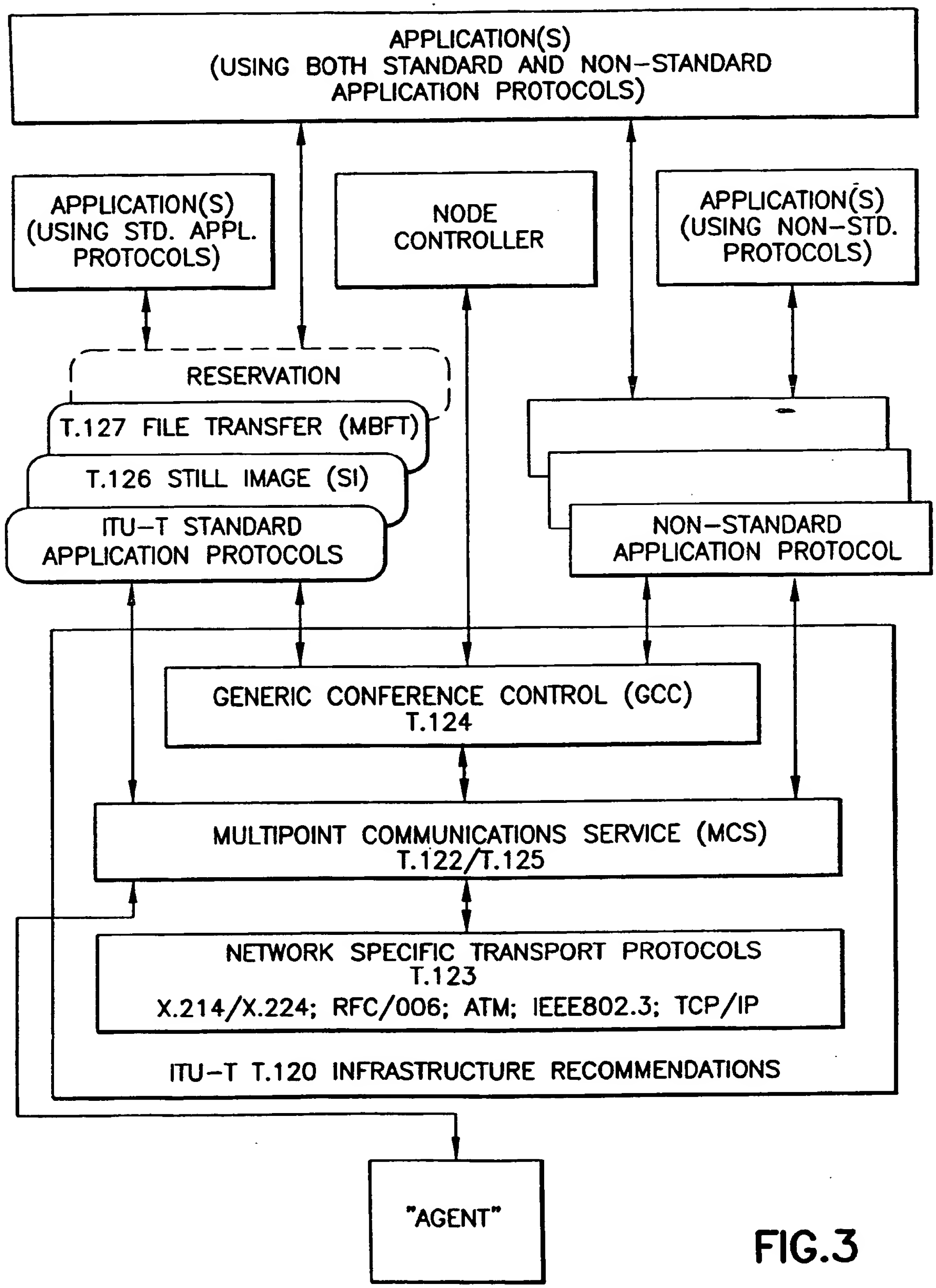
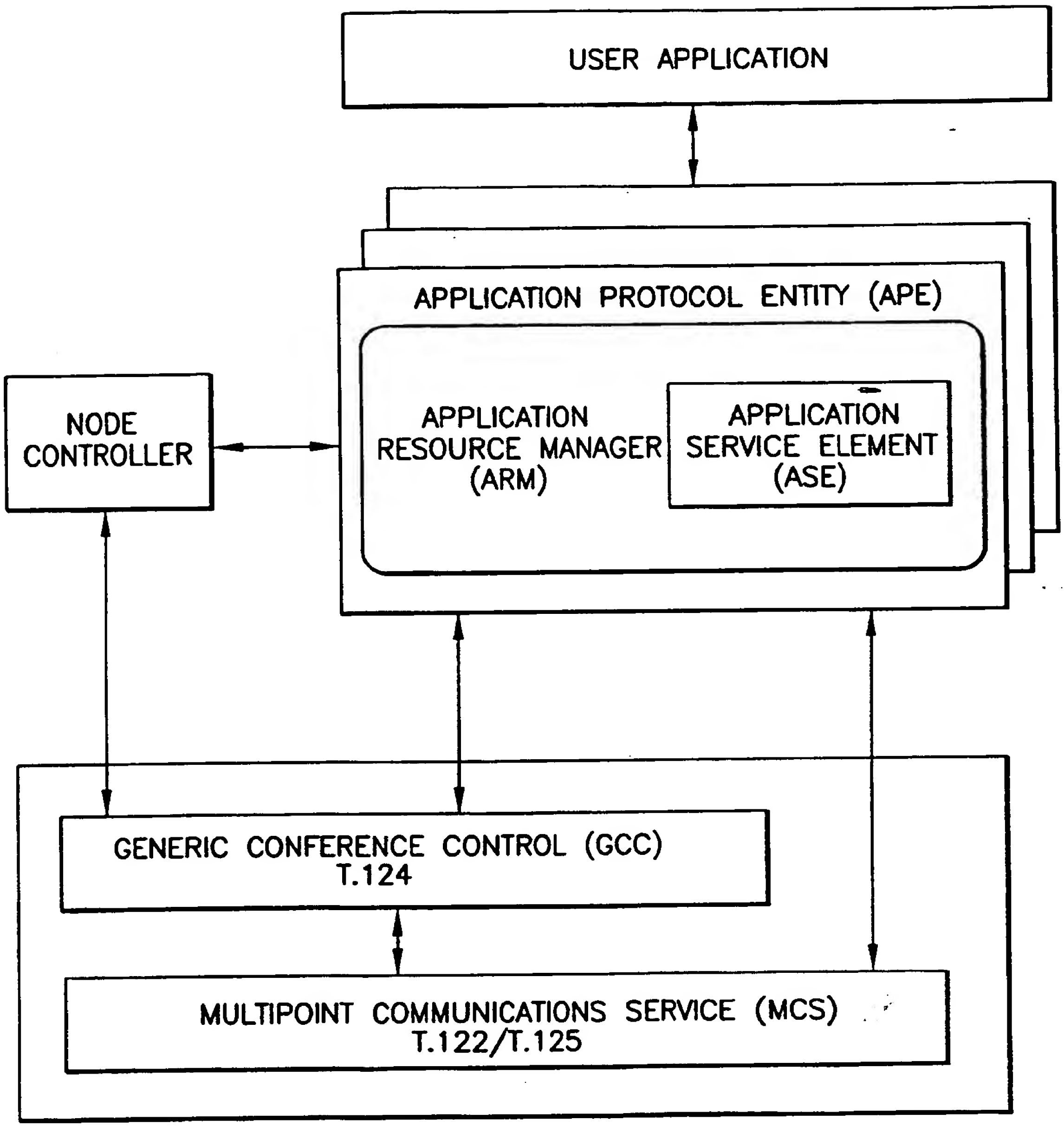

APPLICATION(S)
(USING BOTH STANDARD AND NON-STANDARD APPLICATION PROTOCOLS)
APPLICATION(S) (USING STD. APPL. PROTOCOLS)
NODE CONTROLLER
APPLICATION(S) (USING NON-STD. PROTOCOLS)
RESERVATION
T.127 FILE TRANSFER (MBFT)
T.126 STILL IMAGE (SI)
ITU-T STANDARD APPLICATION PROTOCOLS
NON-STANDARD APPLICATION PROTOCOL
GENERIC CONFERENCE CONTROL (GCC) T.124
MULTIPOINT COMMUNICATIONS SERVICE (MCS) T.122/T.125
NETWORK SPECIFIC TRANSPORT PROTOCOLS T.123
X.214/X.224; RFC/006; ATM; IEEE802.3; TCP/IP
ITU-T T.120 INFRASTRUCTURE RECOMMENDATIONS
"AGENT"

FIG.3



T.120 GENERIC APPLICATION MODEL

FIG.4

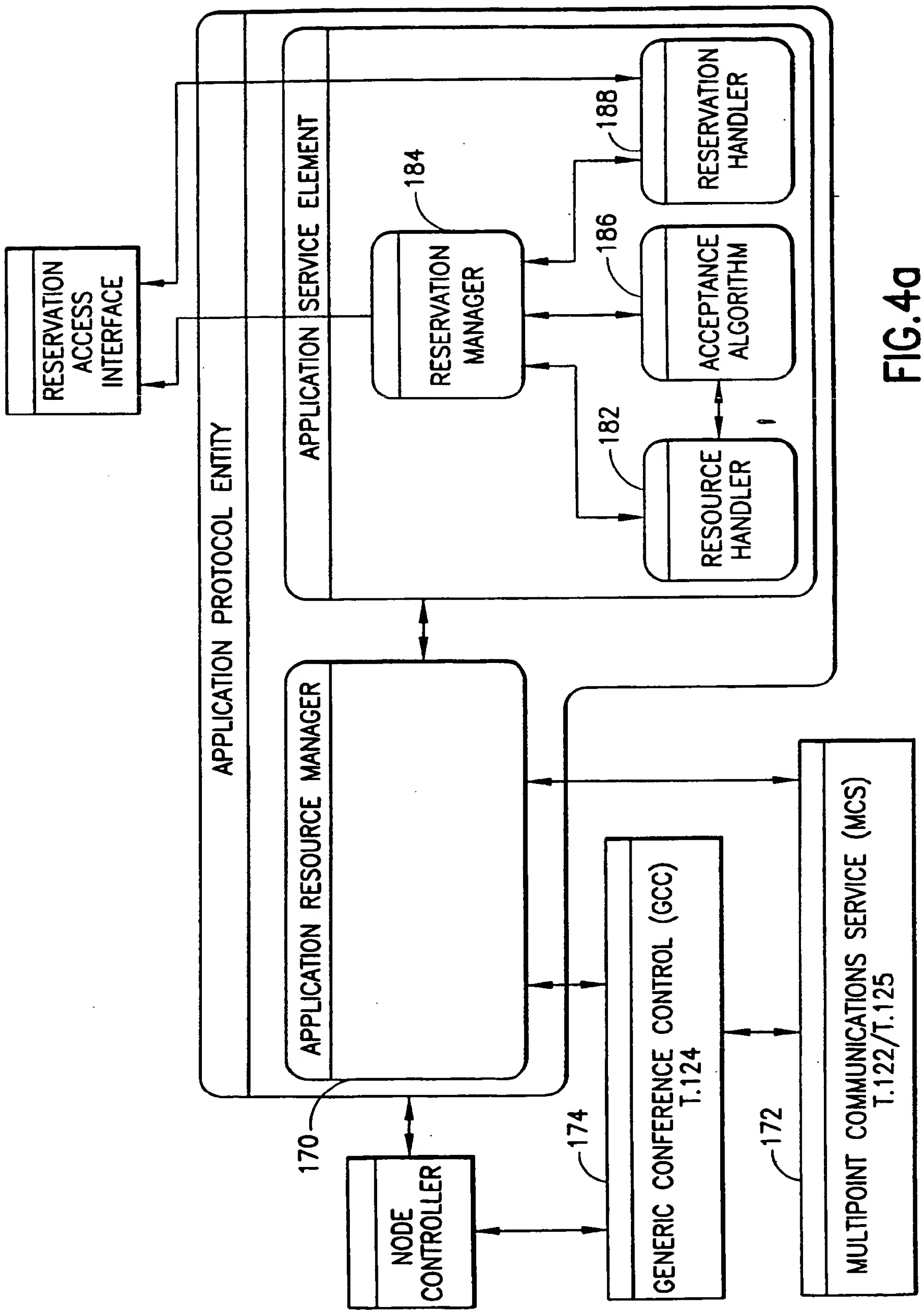

RESERVATION ACCESS INTERFACE
APPLICATION PROTOCOL ENTITY
APPLICATION SERVICE ELEMENT
RESERVATION MANAGER
184
186
188
182
RESERVATION HANDLER
ACCEPTANCE ALGORITHM
RESOURCE HANDLER
APPLICATION RESOURCE MANAGER
170
NODE CONTROLLER
174
GENERIC CONFERENCE CONTROL (GCC) T.124
172
MULTIPOINT COMMUNICATIONS SERVICE (MCS) T.122/T.125

FIG.4a

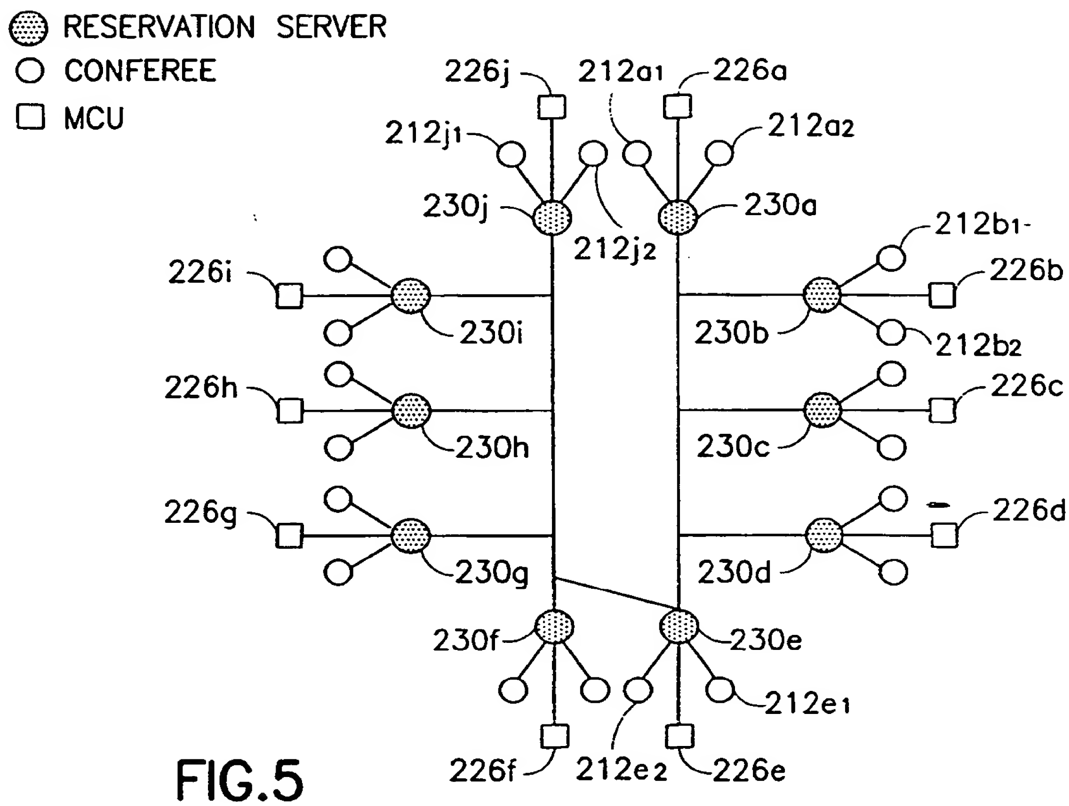

RESERVATION SERVER
CONFEREE
MCU
226j
212a1
226a
212j1
212a2
230j
230a
212j2
212b1
226i
226b
230i
230b
212b2
226h
226c
230h
230c
226g
226d
230g
230d
230f
230e
212e1
226f
212e2
226e

FIG.5

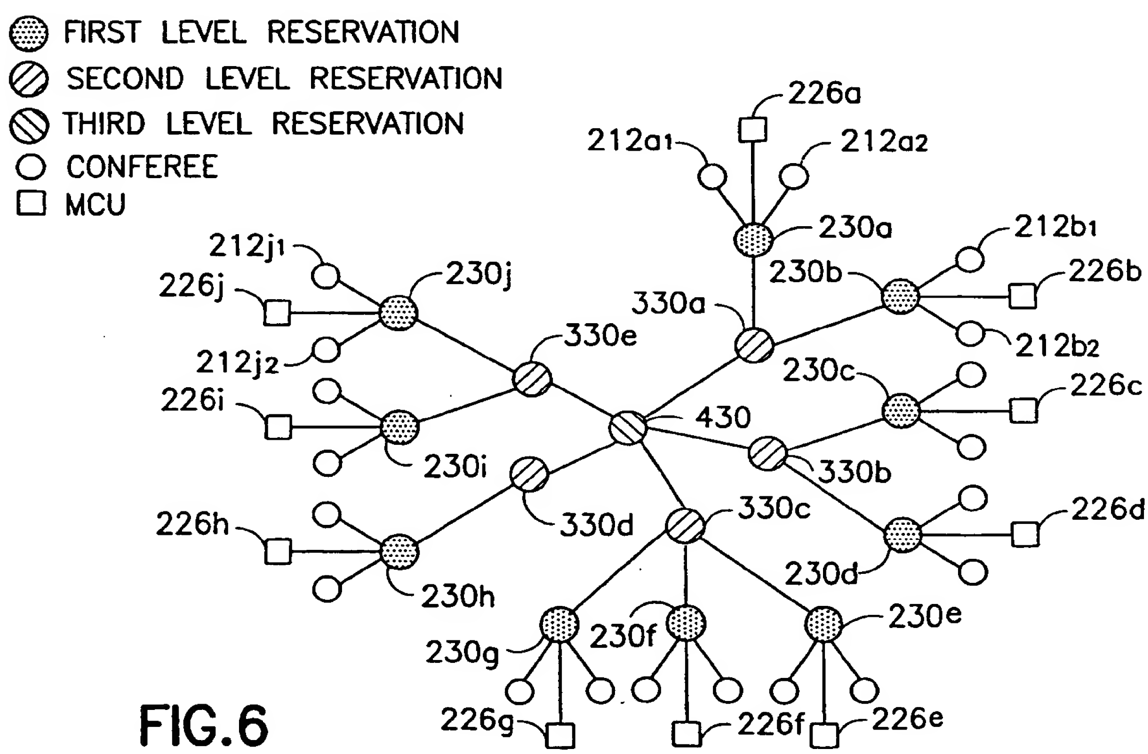

FIRST LEVEL RESERVATION
SECOND LEVEL RESERVATION
THIRD LEVEL RESERVATION
CONFEREE
MCU
226a
212a1
212a2
230a
212b1
212j1
230b
226b
226j
230j
330a
330e
212b2
212j2
230c
226c
226i
430
230i
330b
226d
226h
330d
330c
230d
230h
230e
230f
230g
226g
226f
226e

FIG.6

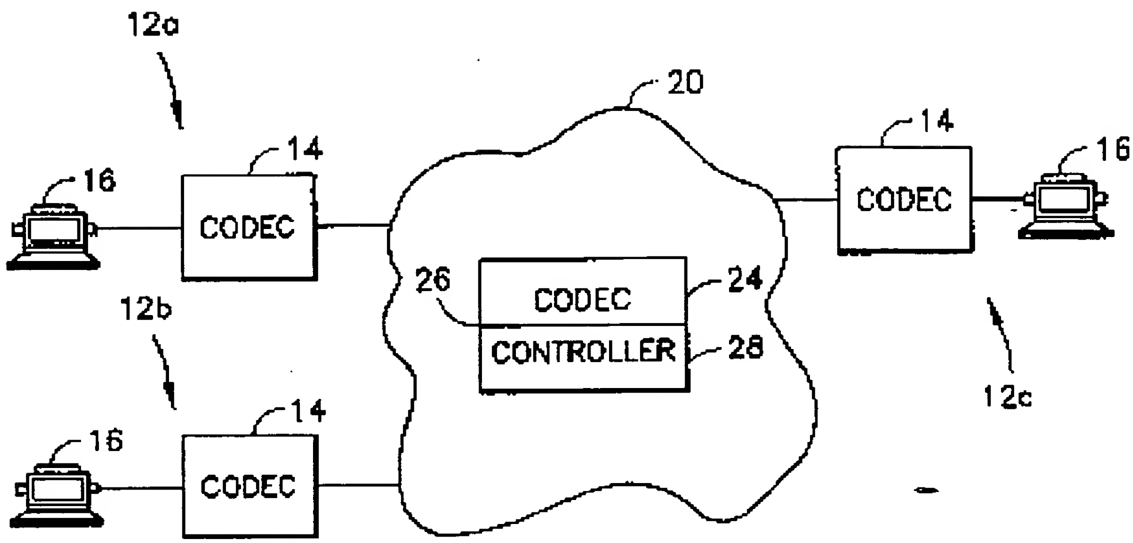


FIG.1
PRIOR ART

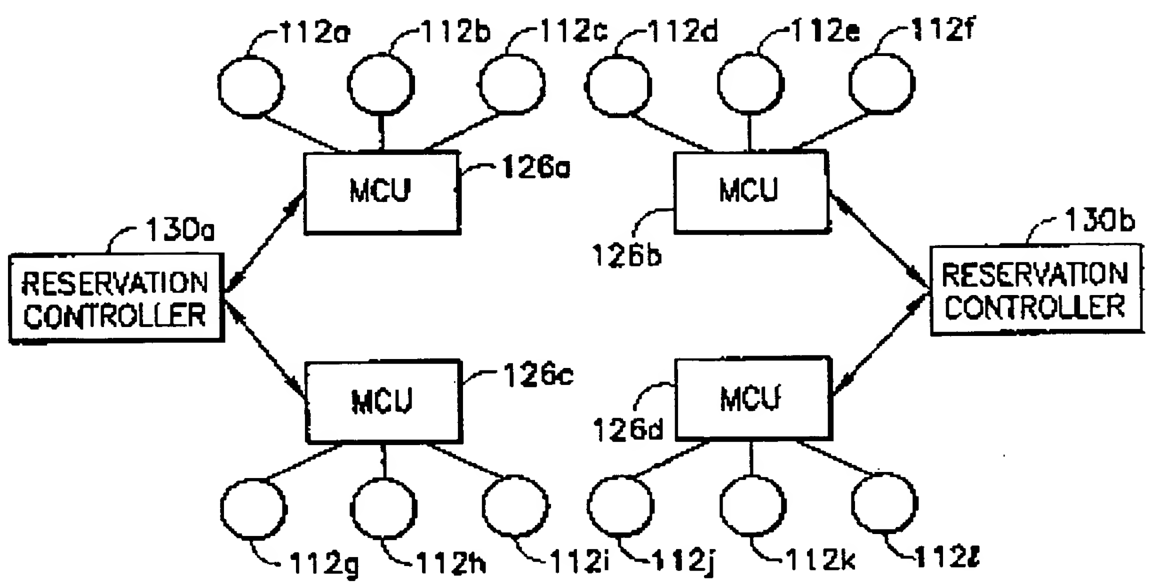


FIG.2

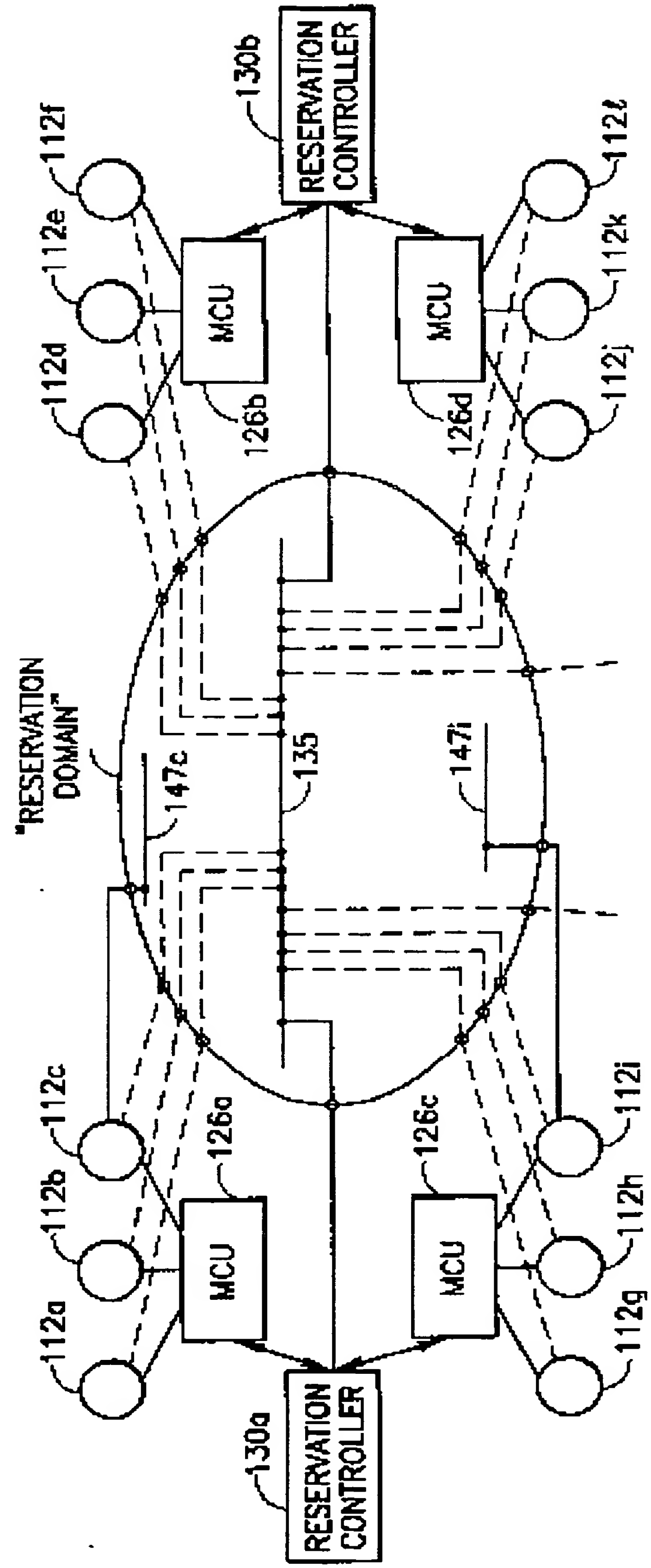

RESERVATION CONTROLLER
130b
112f
112e
112d
MCU
126b
112ℓ
112k
112j
MCU
126d
"RESERVATION DOMAIN"
147c
135
147i
112c
126a
112b
112a
MCU
RESERVATION CONTROLLER
130a
126c
MCU
112i
112h
112g

FIG.2a

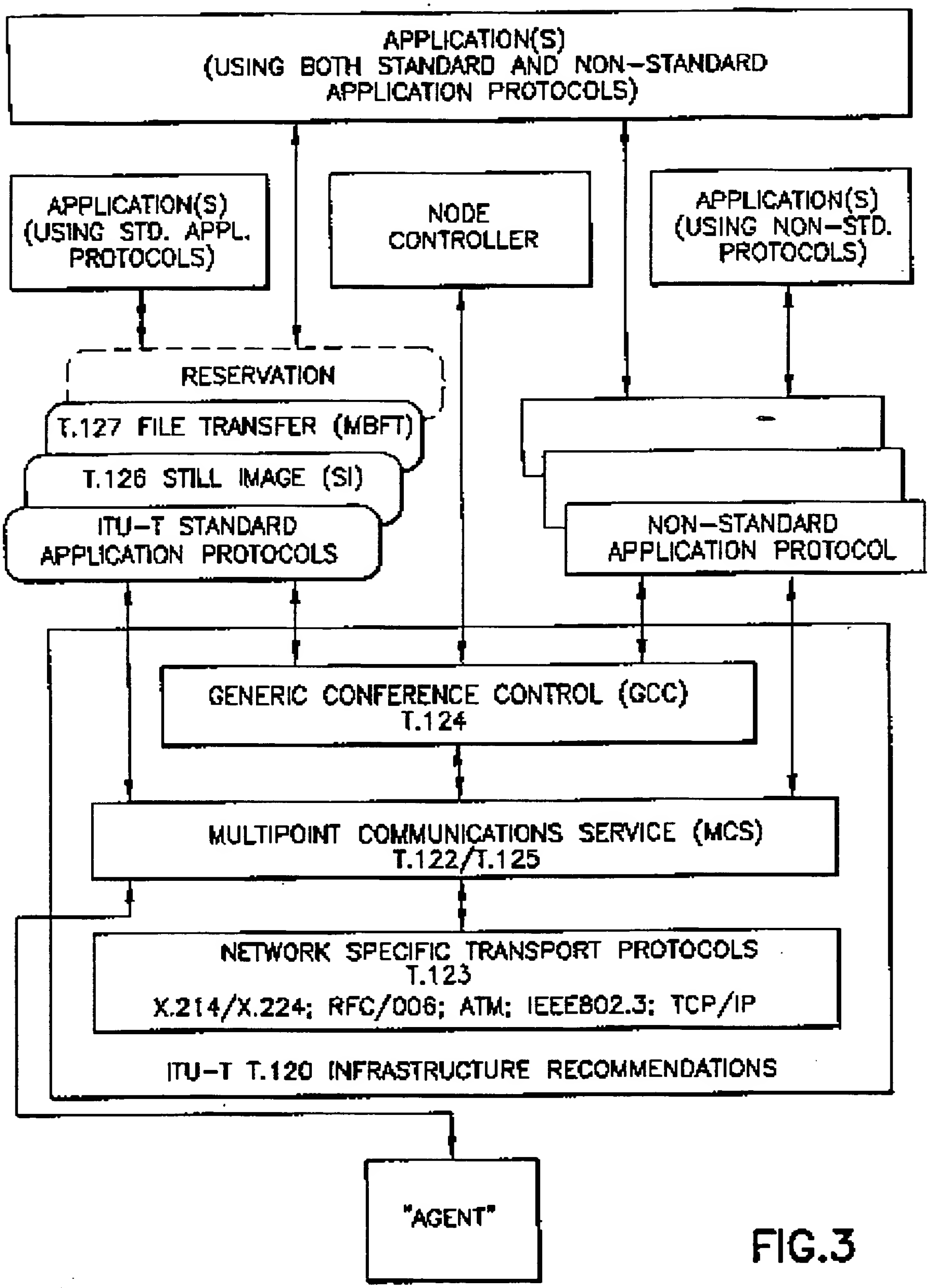
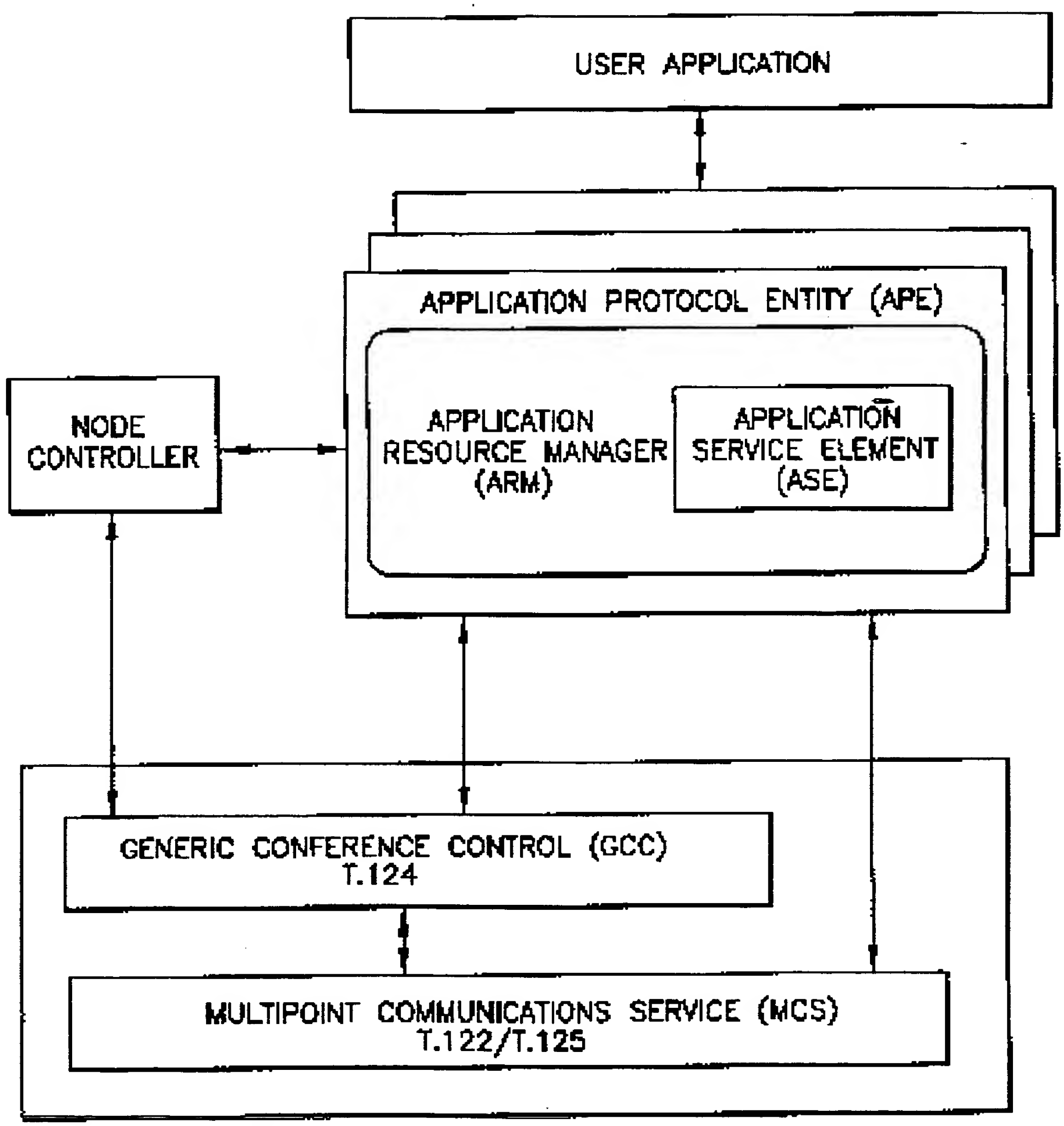
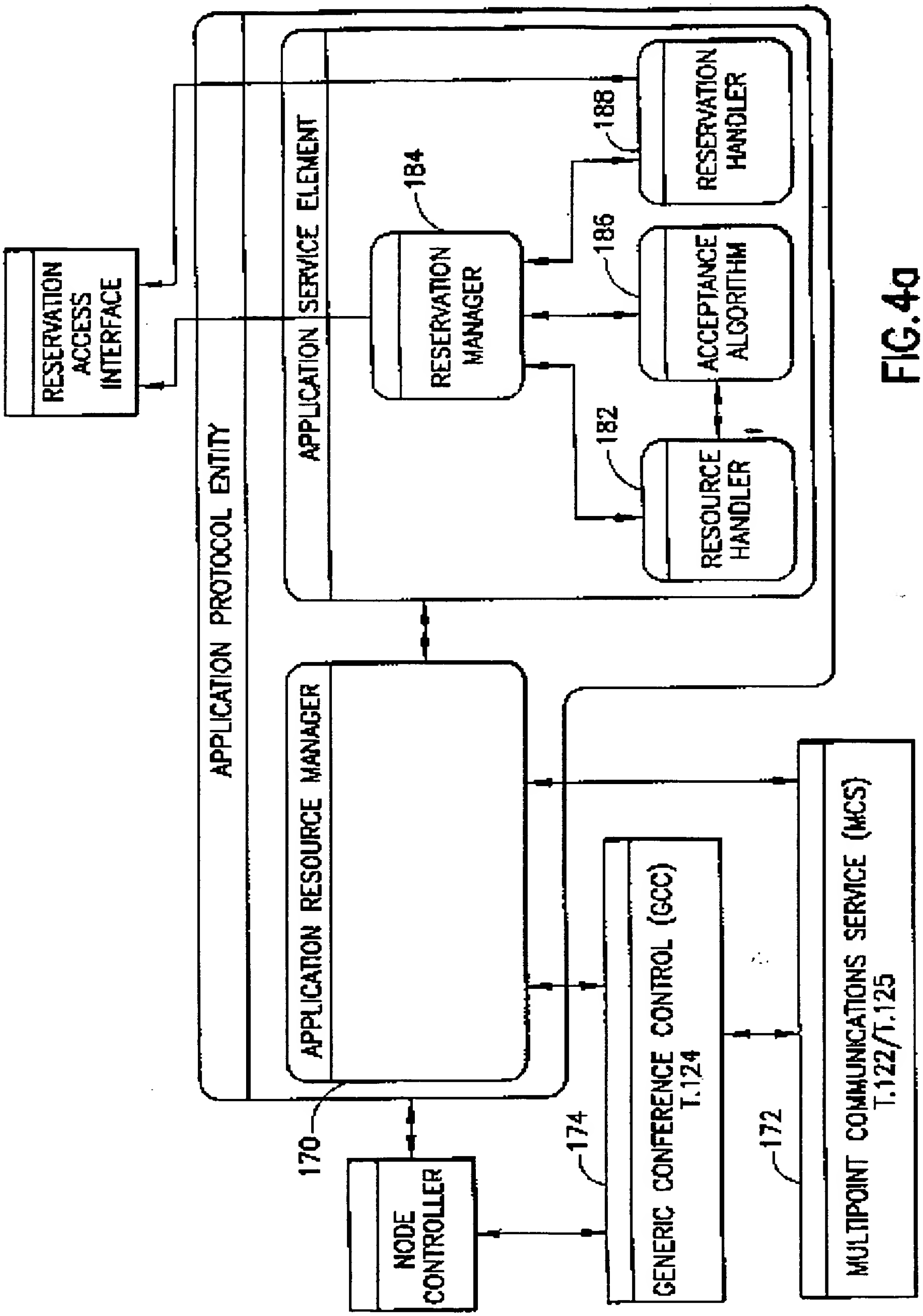

APPLICATION(S)
(USING BOTH STANDARD AND NON-STANDARD
APPLICATION PROTOCOLS)
APPLICATION(S)
(USING STD. APPL.
PROTOCOLS)
NODE
CONTROLLER
APPLICATION(S)
(USING NON-STD.
PROTOCOLS)
RESERVATION
T.127 FILE TRANSFER (MBFT)
T.126 STILL IMAGE (SI)
ITU-T STANDARD
APPLICATION PROTOCOLS
NON-STANDARD
APPLICATION PROTOCOL
GENERIC CONFERENCE CONTROL (GCC)
T.124
MULTIPOINT COMMUNICATIONS SERVICE (MCS)
T.122/T.125
NETWORK SPECIFIC TRANSPORT PROTOCOLS
T.123
X.214/X.224; RFC/006; ATM; IEEE802.3; TCP/IP
ITU-T T.120 INFRASTRUCTURE RECOMMENDATIONS
"AGENT"

FIG.3



T.120 GENERIC APPLICATION MODEL

FIG.4



RESERVATION ACCESS INTERFACE
APPLICATION PROTOCOL ENTITY
APPLICATION SERVICE ELEMENT
184
RESERVATION MANAGER
188
RESERVATION HANDLER
186
ACCEPTANCE ALGORITHM
182
RESOURCE HANDLER
APPLICATION RESOURCE MANAGER
170
NODE CONTROLLER
174
GENERIC CONFERENCE CONTROL (GCC) T.124
172
MULTIPOINT COMMUNICATIONS SERVICE (MCS) T.122/T.125

FIG.4a

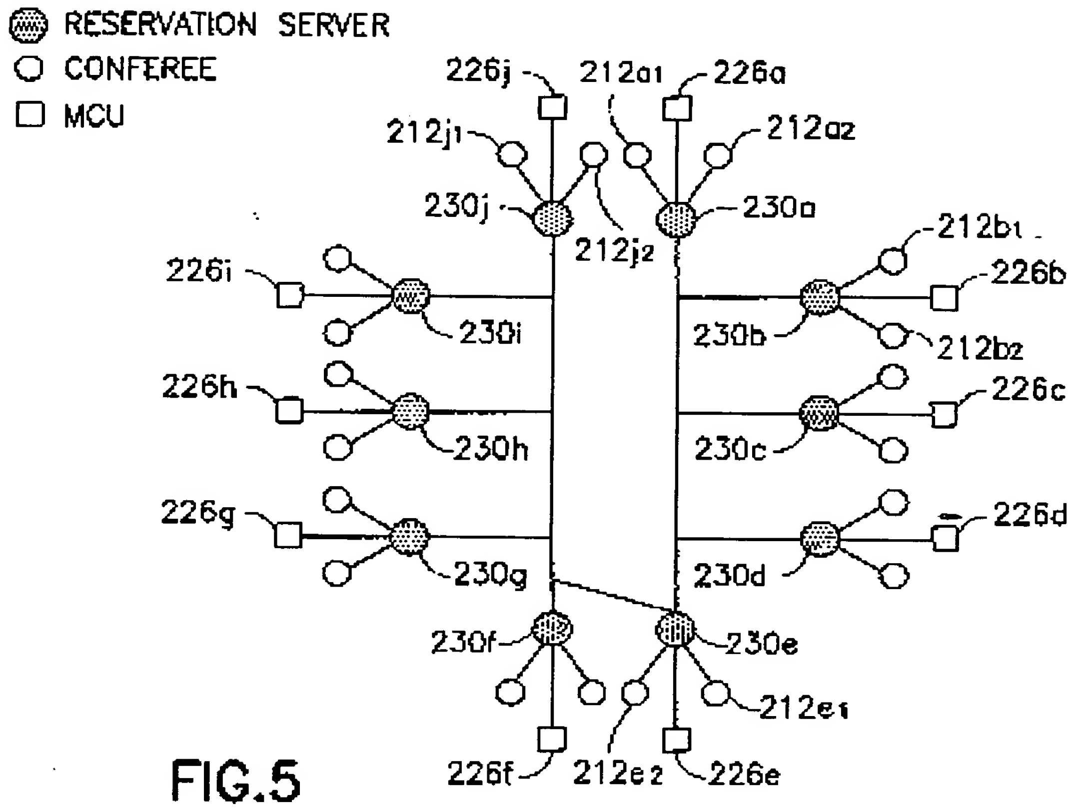

RESERVATION SERVER
CONFEREE
MCU
226j
212a1
226a
212j1
212a2
230j
230a
212j2
212b1
226i
226b
230i
230b
212b2
226h
226c
230h
230c
226g
226d
230g
230d
230f
230e
212e1
226f
212e2
226e

FIG.5

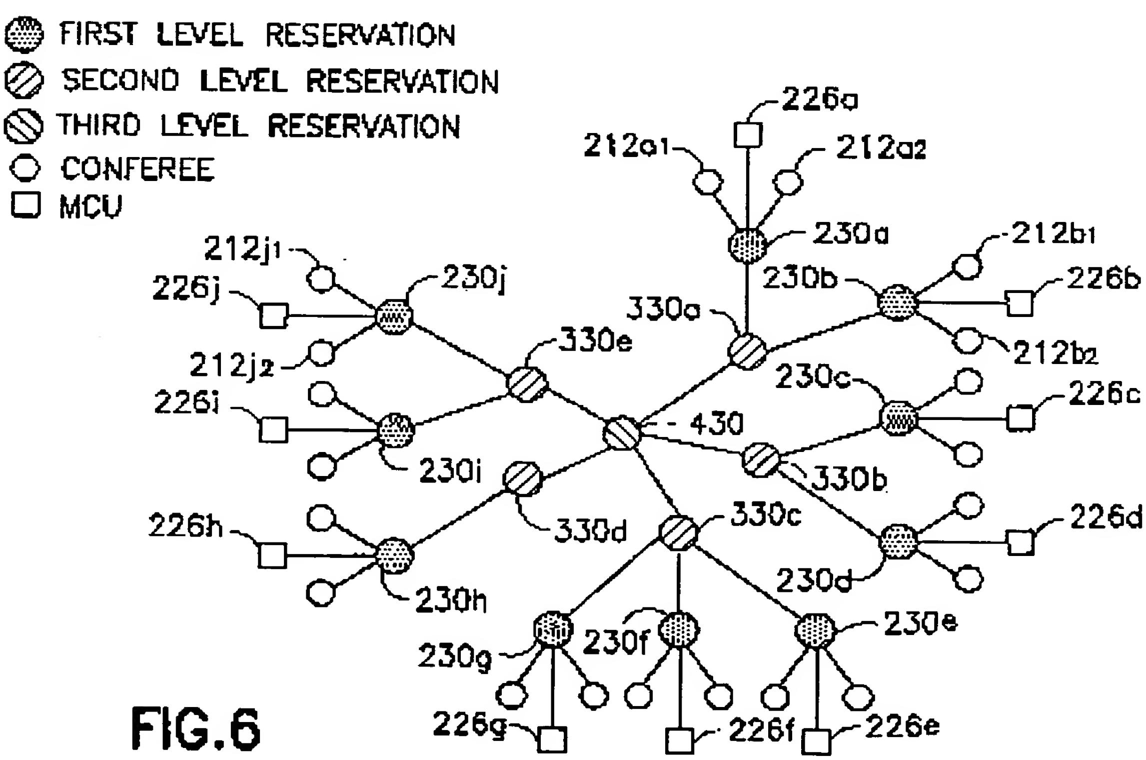

FIRST LEVEL RESERVATION
SECOND LEVEL RESERVATION
THIRD LEVEL RESERVATION
CONFEREE
MCU
226a
212a1
212a2
230a
212b1
212j1
230j
226j
230b
226b
330a
330e
212b2
212j2
230c
226c
226i
430
330b
230i
330d
330c
226d
226h
230h
230d
230e
230g
230f
226g
226f
226e

FIG.6

XP-000700319

ELSEVIER

Computer Networks 31 (1999) 205–223 (19)

COMPUTER NETWORKS

pd. 11/02/99

ITU-T standardization activities for interactive multimedia communications on packet-based networks: H.323 and related recommendations

E

James Toga [a,*], Jörg Ott [b,1]

[a] Intel, JF3-212 2111 N.E 25th Avenue, Hillsboro, OR 97124-5961, USA

[b] University of Bremen, Computer Science Department, Center for Computing Technology, MZH 5180, Bibliothekstr. 1, D-28359 Bremen, Germany

Abstract

The Telecommunication Sector of the International Telecommunication Union (ITU-T) has developed a series of recommendations together comprising the H.323 system that provides for multimedia communications in packet-based (inter)networks. This series of recommendations describe the types and functions of H.323 terminals and other H.323 devices as well as their interactions. The H.323 series of recommendations includes audio, video and data streams, but an H.323 system minimally requires only an audio stream to be supported. Motivated by straightforward interoperability with the ISDN and PSTN networks and a variety of other protocols, the recommendation H.323 has been accepted as being *the* standard for IP telephony, developed by the ITU-T and broadly backed by the industry—which is also adopted by both the Voice over IP (VoIP) forum and the European Telecommunication Standards Institute (ETSI). This paper presents an overview of the H.323 system architecture with all its functional components and protocols and points out all the related specifications. © 1999 Elsevier Science B.V. All rights reserved.

Keywords: Multimedia communication; Teleconferencing; Internet telephony; CSCW; Computer telephony integration (CTI); Mbone; Multicast

1. Introduction

The personal computer and other digital devices are rapidly becoming key communication tools for millions of users worldwide. The importance of digital and data network communications has greatly increased with the explosion of the Internet. While electronic mail is still the dominant method of interactive computer communications, electronic conferencing and IP-based telephony are becoming increasingly attractive. The adoption of packet switching and its merging with circuit switching, helps drive this communications migration. There are many reasons for this, among them pricing advantages due to improved resource utilization, seamless transitions between monomedia and multimedia communications, as well as between human-to-computer (e.g. web-based) and interpersonal interactions. Additional motivations exist such as advanced and flexible fea-

* Corresponding author. Tel.: +1-503-2648816; fax: +1-503-2643485; e-mail: jtoga@ibeam.intel.com.

[1] Tel.: +49-421-201-7028; Fax: +49-421-218-7000; E-mail: jo@tzi.uni-bremen.de.

PII: S0169-7552(98)00267-0

tures that may be offered as inherent part of the system (rather than as complex and expensive add-ons); and the ultimate integration of voice and data networks and systems. Ubiquitous packet based, real-time communication offers many challenges: with respect to technical complexity and particularly in terms of deployment and (organizational) integration. One of the key issues related to the success of digital and computer communications is a *standard* way of providing connectivity—from call control (finding other parties, ringing, etc.) to media encoding to administrative controls (admission control, billing, etc.). Standards for real-time multimedia communications such as H.323 provide the foundation for global interoperability and thus enable future connectivity expansion from a technical as well as from an economic point of view.

For interactive multimedia communications on packet-based networks including IP-based telephony, the relevant standard of the Telecommunication Sector of the International Organization for Standardization (ITU-T) is the H.323 series of recommendations [2] comprising besides H.323 [4] itself H.225.0 (core message definitions) [1], H.245 (media channel control) [3], H.235 (security framework) [2], H.450.x (supplementary services) [6], and H.332 (extensions for large group conferences) [5] [3]. The initial version of H.323 containing the base functionality for IP-based multimedia communications was ratified in summer 1996 after one year of intense development efforts. This version provided a convergence point for the industry and prevented the development of a variety of incompatible products on a large scale. The H.323 protocol was developed by utilizing or taking into account existing technology where possible and appropriate: RTP/RTCP, and standard codecs were re-used without change; H.323 and H.245 were enhanced to include hooks to make use of existing means for achieving Quality of Service (QoS) [4]. Only where no applicable solutions existed, new protocols were developed. In essence, this applies only to policy control and management functionality; allowing network administrators to control (network) resource utilization by H.323 components. During the most recent cycle in the ITU-T standardization, a number of enhancements to H.323 and its related protocols resulted in the 1998 version, manifested as revisions to H.323, H.225.0, and H.245 in addition to new related recommendations (H.235, H.332, H.450.x). These new features satisfy demands for new functionality and extensions to existing services. Many of them stem from a broadened scope with the most important focus—IP telephony—motivated by the increased commercial use of H.323 for this environment.

This paper is organized as follows: Sections 2–5 address the technical foundation based upon the initial 1996 recommendations. Section 2 outlines the functionality offered by H.323 and presents its architecture. Sections 3–5 provide details about the H.323 system components, its protocols, and the operational procedures, respectively. Following this, Section 6 explores the most important extensions of H.323 version 2 including enhanced support for IP telephony, security functions, and large group conferences, and also briefly addresses on-going work. Section 7 concludes this paper with a brief evaluation of the status of H.323.

2. Overview of the H.323 system

The H.323 series of recommendations describes systems, logical components, messages and procedures that enable real-time, multimedia calls to be established between two or more parties on a packet network. This section first outlines the services provided by a H.323 system and then defines the scope

[2] In ITU-T language, the H.323 standard is formally referred to as a *Recommendation*.

[3] Work is continuing and new functionality is being added—as new recommendations or additions to existing ones—while this article is being prepared. These additions comprise further supplementary services, definition of Management Information Bases (MIBs), operation of H.323-based facsimile systems among many other enhancements. As those are not mature at the time of writing they cannot be addressed in this article.

[4] The H.245 protocol provides QoS capability signaling (including specific parameters from RSVP) and the opening of media channels can request RSVP reservation modes in conjunction with the RTP streams. Additionally, Appendix II of H.323 presents a profile for use with RSVP.

of the H.323 series of recommendations. The latter includes a brief introduction to all the system and protocol components of H.323 and their purpose in the system.

2.1. H.323 services

H.323 is designed to extend traditionally circuit-based audiovisual and multimedia conferencing services into packet (i.e. IP-based) corporate networks. The voice-only subset of H.323 provides the platform for IP-based telephony. In both areas, seamless interoperation with circuit-switched networks (ISDN, PSTN) as well as provision of well-known conferencing and PBX services are achieved by H.323; as is the straightforward extensibility to include novel features.

The H.323 system aggregates a number of standards, which together allow establishing and controlling point-to-point calls as well as multipoint conferences. Personal computers and other devices—regardless of the hardware, operating system, and software employed—can inter-operate sharing a rich mixture of audio, video, and data across all forms of packet-based networks (intranets as well as the Internet). Seamless interoperation with systems on circuit-switched networks is supported via Gateways. H.323 provides a tightly controlled communications model, with explicit control and media connections set up between participants. Media transmission may occur point-to-point via unicast or take advantage of multicast capabilities of the underlying networks. The selection of available media, their respective formats, and the transmission topology are dynamically negotiated. In addition to interactive multimedia conferencing, H.323 also has specific provisions for other forms of communication—that are either special cases and/or may be part of/extensions to multi-media conferences —, such as multimedia streaming, distance learning, and IP telephony. As each of these models of communication coalesces in a different manner, H.323 enables both "join" and "invite" modes in establishing communications. Finally, H.323 defines mechanisms to integrate directory functions, admission control, and call routing that allow implementations (and eventually administrators/users) to define virtually arbitrary usage policies for the H.323 environment.

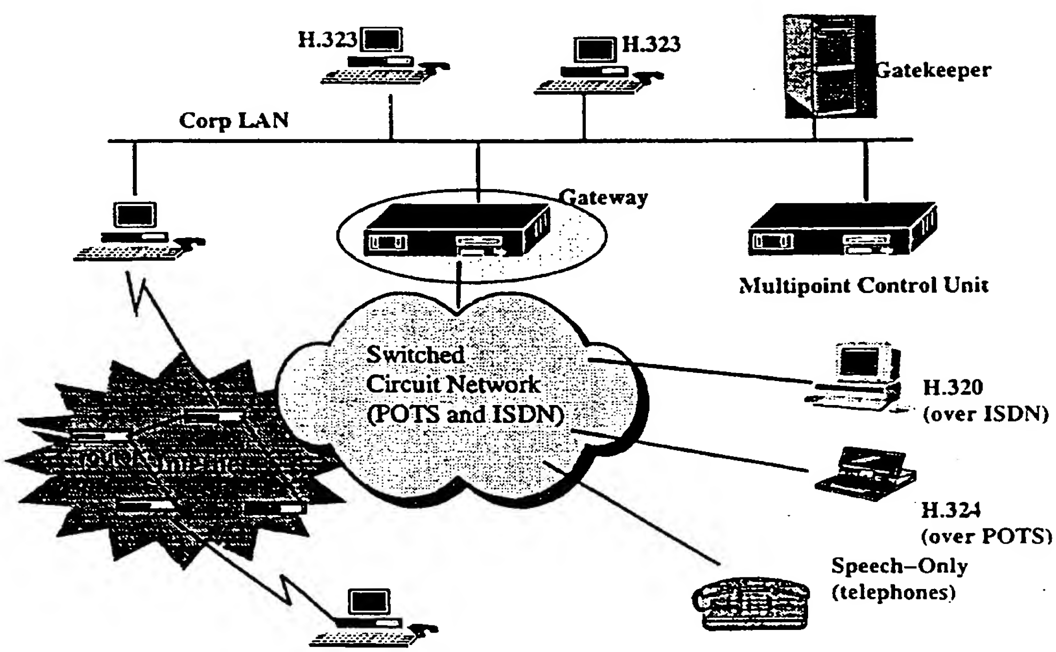


Fig. 1. Environment of H.323 and sample network topology.

2.2. Scope of H.323

Although H.323 is minimally defined to operate utilizing only peer H.323 terminals, the recommendation defines a number of additional logical H.323 elements. These elements include Gatekeepers for policy control and address resolution; Multipoint Controllers (MCs) and Multipoint Processors (MPs) —both of which may be combined to form a Multipoint Control Units (MCU)—for multiparty conferencing; as well as Gateways and Proxies for operation across network boundaries. The elements are defined in terms of specific logical functions and protocol responsibilities; there are no preconditions on the physical location or combination of elements in a network. Although H.323 clearly defines services and interactions between all of these logical elements, there are no specific hardware or software requirements mandated. Fig. 1 depicts the environment of H.323 in terms of the logical system components and also shows a sample network topology indicating a variety of interactions covered by H.323 [4].

Fig. 2 illustrates the block diagram of a generic H.323 endpoint showing all the core protocols. Contained within the large light gray block in the center are those protocols within the scope of the H.323 series of recommendations. The darker shaded blocks on the left of the figure contain application components that may be different for each implementation. On the right side of the figure is the generic packet network interface—while H.323 is defined to allow implementation on arbitrary (connectionless) packet-switched networks (including IP, IPX, and others), only IP networks are of any relevance in practice. While definition of the network and transport protocols themselves are outside the scope of the recommendation, H.323 precisely specifies the requirements on those protocols: provision of a reliable connection-oriented (e.g. TCP) along with an unreliable connectionless (e.g. UDP) mode of operation. For certain functions, H.323 assumes the IP multicast service model for the unreliable transport. The protocol components indicated by the white boxes in Fig. 2 provide:

- call admission and address resolution mechanisms, including call routing (admission control, H.225.0),
- call establishment and termination (call control, H.225.0),

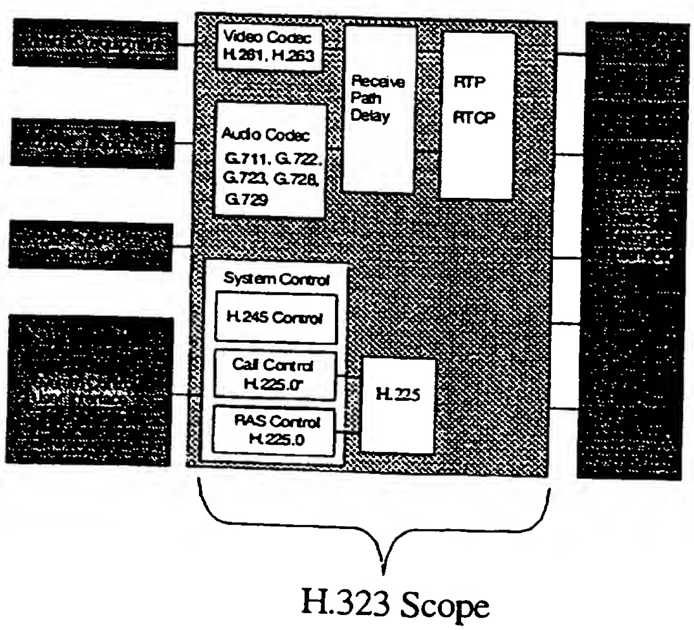


Fig. 2. H.323 core protocols.

- capability negotiation and media channel establishment (H.245), and
- runtime media transport and control signalling (RTP/RTCP).

The following section outlines the various logical elements of the H.323 system and their respective roles. A more detailed description of the H.323 core protocols is given in Section 4. Then, Section 5 gives an overview of the operation of an H.323 system by outlining interactions between H.323 elements and the interaction of the various protocols.

3. H.323 elements

This section describes the logical elements that operate in the H.323 environment [4]. Four main elements are defined: terminals, Gatekeepers, Gateways and Multipoint Control Units (consisting of Multipoint Controllers and Multipoint Processors). An H.323 Proxy is a fifth component that may be transparent to H.323 protocol operation; it is not explicitly covered in an ITU-T recommendation. The synopsis of their function is:

- Terminal – what humans utilize in a conference (e.g. a PC or a phone),
- Gateway (GW) – bridging to other network environments,
- Multipoint Controller (MC) – coordinated control for multiparty conferences,
- Multipoint Processor (MP) – audio and video mixing or switching,
- Multipoint Control Unit (MCU) – contains MC, MP, and optionally a T.120 MCU,
- Gatekeeper (GK) – (administrative) control and "call routing", and
- H.323 Proxy – controls how H.323 conferences may transit firewalls.

These H.323 elements are described in more detail in the remainder of this section.

3.1. Terminal

Terminals together with Gateways and MCUs are collectively referred to as endpoints. A terminal is typically the one element that exists in all H.323 usage scenarios. It is the terminal which generates and ultimately receives H.323 calls or participates in a multi-point conference. This device may be anything from a simple telephone-like box to a high-end computer workstation. All terminals must implement audio communications (at minimum, in accordance with the mandatory audio codec G.711) with support for video and data being optional. All terminals must implement the H.225.0 call control (derived from Q.931) and the H.225.0 admission control (Registration, Admission, and Status – RAS) protocols for call and conference establishment along with the H.245 protocol for capability and media stream control.

3.2. Gateway

A Gateway provides the ability for H.323 devices to interoperate with other devices in heterogeneous (e.g. non-H.323-based) network environments. Besides the underlying network/transport mechanisms (e.g. ISDN, PSTN), these environments can also be different with respect to the communication protocols used, the media encoding employed, etc. Consequently, an H.323 Gateway maps call control protocols (e.g. Q.931 as found in ISDN to H.225.0), control protocols (e.g. H.242 as found in H.320 systems to H.245), media encoding (e.g. G.711 in ISDN to G.723.1), and media serialization (e.g. octet framing of ISDN to RTP packetization). H.323 Gateway procedures specify, among many other details, how incoming and outgoing calls are to be handled, how two-stage dialing works, when call establishment completes, from which point in time media flow is possible, and how a call is terminated. The H.323 standard defines a number of Gateway devices currently including Gateways for H.320 (ISDN-based video conferencing terminals), for H.324 (PSTN-based video conferencing terminals), and Plain Old Telephone System (POTS, PSTN) devices. This list will expand, as Gateways are developed to bridge to other environments.

3.3. Multipoint control and processing elements

A Multipoint Control Unit (MCU) provides the ability to hold multiparty, multimedia conferences. It coordinates all of the media capabilities of the participants and may provide features such as audio mixing and video selection for endpoints that cannot

accomplish this locally as well as transcoding of media streams to bridge between otherwise incompatible endpoints. Furthermore, an MCU may provide chair control and conference roster capabilities in multi-point conferences. It also facilitates the graceful entrance and exit of conference participants. In the telephony environment, some PBX supported functions of an audio "bridge" might be considered analogous to an MCU. H.323 refines the standard definition of an MCU drawn from H.320 systems, by creating two logical elements: a multi-point controller (MC) and a multi-point processor (MP). The MC provides the call control coordination needed in a multi-point conference if the media mixing and selection can be performed by the individual participants. The MP component provides the audio mixing, the video mixing or selection, and the handling of (T.120-based [22]) multipoint data communications, and may also perform transcoding of media streams.

3.4. Gatekeeper

Regions of an IP-based network (such as topologically adjacent ones) are grouped into zones for administrative purposes. A Gatekeeper administers each zone. The Gatekeeper acts as monitor of all H.323 calls within its zone on the network and provides two main services: call admission and address resolution.

All endpoints register with their Gatekeeper prior to performing any further H.323-related action. An H.323 client that wants to place a call, does so with the assistance of the Gatekeeper. The Gatekeeper provides the address resolution from an *alias* name to a specific transport address of the destination client during the initial Admission Request (ARQ) signalling. Note that the means the Gatekeeper chooses to perform this address translation—lookup in its own registration tables, query of directory server via the Lightweight Directory Access Protocol (LDAP), invocation of any proprietary user location protocols, etc.—are deliberately left unspecified in H.323.

During this address resolution phase, the Gatekeeper may also make permission decisions based upon available bandwidth or any other policy such as identity of the caller, or priority of other network functions. The Gatekeeper can act as an administration point on the network for IT/IS managers to control H.323 traffic on and off of the network (share of available bandwidth allocated to H.323 multimedia traffic), utilization of shared resources (such as MCUs), or access to "external lines" via Gateways. The Gatekeeper may also provide advanced features for routing calls to specific Gateways or extended telephony-like services such as call status, call accounting and PBX-like features—a prerequisite for this is that the Gatekeeper receives, processes, optionally responds to, and/or forwards call control messages exchanged between the endpoints (Gatekeeper-routed call model, refer to Section 5.3). The Gatekeeper is not a required element in an H.323 environment, i.e. network administrators may choose to run H.323 without a Gatekeeper; but in this case, the endpoints must have other means for determining the transport address of the other endpoint(s) being called. Gatekeepers are required to implement the RAS protocol from H.225.0 and may optionally implement the H.225.0 call control and H.245 protocols if they are to supply advanced services. Services such as call path provisioning (i.e. finding an unloaded Gateway) or call management (i.e. activating an MCU in a call) may be provided in this fashion.

3.5. Proxy

An H.323 Proxy acts in a manner similar to other types of proxies: it acts on behalf of elements on one side to contact elements on the other. H.323 Proxies must fulfill many of the requirements of an H.323 Gateway and provide the same interfaces and functions that a Gateway presents. In practice, H.323 Proxies are typically co-located with an enterprise firewalls or Gatekeepers and monitors all H.323 calls between the enterprise and the Internet [5]. The Proxy

[5] Note that a Proxy operating in an H.323 environment may (but need not) be *explicitly* detected and used by an endpoint; however, protocol exchanges are not modified. Additional addressing information *may* be presented to the Proxy but, in general, the endpoints do not change their behavior. Some implementations place the proxy behind a Gatekeeper thereby insulating any H.323 entities from its presence (assuming the Gatekeeper-routed call model, see Section 5.3).

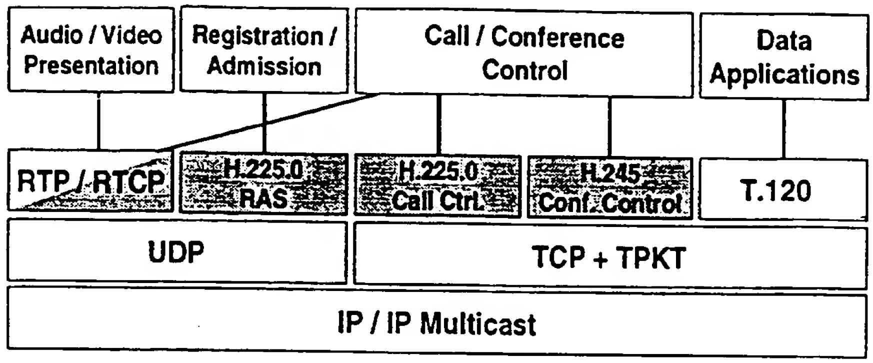


Fig. 3. H.323 protocol stack.

ensures that only valid H.323 traffic goes through the firewall. It also enforces access control policies for users on either side of the Proxy (these are different from the bandwidth controls of the Gatekeeper). Access control policies may include determining which users can initiate or receive H.323 calls, what destinations are appropriate, and whether a particular user is allowed to use video facilities.

4. H.323 Protocol components

Fig. 3 outlines the protocol hierarchies of H.323 on top of an IP-based network. The shaded elements indicate the protocols defined within the scope of H.323. The uppermost layer indicates the (application system) functions for which the respective protocols are used. Both the H.225.0 call signalling and the media control (H.245) depend on a reliable transport and hence are carried in TCP connections, the H.225.0 RAS channel uses UDP as transport layer, and the audio/video streams use RTP on top of UDP. Real-time media streams may be encoded following the ITU standard voice and video codecs (G.7xx and H.26x, respectively), using codecs from other organizations (e.g. GSM defined by ETSI), or proprietary codecs.

4.1. H.225.0: Call admission and call control

The H.225.0 document [1] contains the definitions of all messages exclusively used by H.323 components and required for basic operation of the H.323 system; messages shared with other H.3xx series recommendations (such as H.245 media channel control) and messages providing non-core functionality (such as H.450.x supplementary services) are specified in separate documents (and are discussed subsequently). The H.225.0 document embodies two subprotocols: Registration, Admission, and Status (RAS) and the call control messages derived from Q.931 [6] [7]. It also includes a normative annex, which describes the use of RTP/RTCP in the context of H.323. In general, H.225.0 covers the call setup and the initial call signalling.

4.1.1. RAS channel: registration, admission, and status

The RAS messages are primarily used between the endpoints (terminals, Gateways, MCUs) and their respective Gatekeepers. RAS comprises a number of request/response messages, which facilitate Gatekeeper discovery, endpoint registration, and call activity as signalled to a Gatekeeper. After initial discovery of and registration with their respective Gatekeepers, endpoints use RAS messages to coordinate activities that may change their utilization of Gatekeeper-supervised resources—primarily network bandwidth and shared equipment such as Gateways. Endpoints inquire for permission to increase resource utilization and provide notifications about reduction/termination of resource usage. In addition,

[6] Note that references to Q.931 in this article indicate the signaling as modified by H.225.0, not the text as referenced in [7].

the Gatekeepers use RAS messages to actively query endpoints for their current status (to determine availability of Gateways, to detect silent failures of endpoints, etc.). Thus, the RAS channel puts the Gatekeeper in control of its zone of the network and all its associated resources thereby allowing access policies to be easily defined by the network administrators. Listed below are the RAS messages defined in H.323 version 1 and their intended usage. In general, all request messages are of the form xRQ, with the confirmation or rejection following the form of xCF and xRJ, respectively.

The RAS messages flow on UDP, thus requiring the sequencing and retry mechanisms described in H.225.0. (See Table 1.) An identifier called the Call Reference Value (*CRV*) is included in all of the RAS PDUs to correlate all of the messages that are associated with a particular call. If no Gatekeeper is present in the system—which is determined by the endpoints when they unsuccessfully attempt to discover and register with a Gatekeeper—these messages are not utilized. In the absence of a Gatekeeper it is assumed that address resolution is gained via some mechanism outside the scope of H.323 and that some (potentially non-standard) separate entity is available to police resource utilization (if any policing is needed).

4.1.2. Q.931-based call signalling channel

The Q.931 derived messages may look familiar to those that understand the ISDN signalling of the same name. The Q.931 messages (and procedures) have been modified for use by H.323: the meaning of the original Q.931 header fields is adapted to H.323 and additional H.323-specific information is contained in the *User-User Information Element* (UUIE). All of these messages are exchanged on a reliable connection which simplifies the error handling and sequencing at the expense of setting up a TCP connection.

The Q.931 messages provide the signalling of call setup requests from caller to callee, intermediate signalling (such as indications that a call request is being processed further, the other endpoint is "ringing", etc.) as well as final response(s) from the caller back to the caller. Included in the set of final response messages are the standard acceptance message, call rejection or redirection indications with appropriate reason codes. Additionally the messages may include means for the invocation of other supplementary services known from the telephone world (defined in H.450.x, see Section 6 below). In most simple call scenarios, once the call connection is established, the Q.931 exchanges become dormant and the associated TCP connection may be closed—unless a supplementary service feature is to be invoked later during the call; in this case, the TCP connection may also be re-connected by either endpoint, at the expense of additional signalling and latency though.

4.2. H.245: media and conference control

H.245 [3] is the media control protocol that H.323 system utilizes after the call establishment has completed. The addressing information required to create the separate H.245 protocol channel is passed in the call control message during the Q.931 call establishment phase. H.245 is used to negotiate and establish

Table 1
Overview of H.225.0 RAS messages and their abbreviations

Message function	Request	Confirmation/response	Reject
Gatekeeper discovery	Gatekeeper request (GRQ)	Gatekeeper confirm (GCF)	Gatekeeper reject (GRJ)
Endpoint registration	Registration request (RRQ)	Registration confirm (RCF)	Registration reject (RRJ)
Call admission	Admission request (ARQ)	Admission confirm (ACF)	Admission reject (ARJ)
Media bandwidth control	Bandwidth request (BRQ)	Bandwidth confirm (BCF)	Bandwidth reject (BRJ)
Endpoint/gatekeeper location	Location request (LRQ)	Location confirm (LCF)	Location reject (LRJ)
Status information	Information request (IRQ)	Information response (IRR)	-
Disengage From Call	Disengage request (DRQ)	Disengage confirm (DCF)	Disengage reject (DRJ)
Message not understood	-	Unknown message response (XRS)	-

all of the media channels carried by RTP/RTCP. The H.245 protocol forms the common basis for media and conference control for a number of ITU-T multimedia communication systems including those that operate on a circuit-based transport; thus it contains many messages and procedures not used by H.323 as well as some extensions specific to H.323.

The functionality offered by H.245 that is used by H.323 falls into four categories, the first three of which are mandatory for H.323 operation:

- **Master-slave determination**: to provide a means for tie-breaking in race conditions and to establish an entity (the Multipoint Controller, MC) responsible for central control in case a call is extended to a conference.
- **Capability exchange**: used by H.323 elements to negotiate a common set of operational capabilities. The capability sets describe all aspects of operation between communicating elements: the types of media, number of simultaneous channels, maximum bit-rates, and other options. The capability exchange may occur at any time during a call, allowing for renegotiations of operating characteristics (i.e. bandwidth utilization or processing load change).
- **Media channel control**: After conference endpoints have exchanged capabilities, they may open and close logical channels of media. Logical channels are identifiers used within H.245 as an abstraction for media streams. Flow/rate control and changing of operating modes along with other messages always reference a logical channel. The transmitter of media is limited to opening logical channels that are within the capability set of the receiver. Any audio (and optionally video) are logically *uni-directional* channels. This means that each transmitter is required to open a channel to the recipient(s), implicitly allowing asymmetric use of codecs and different numbers of media flows in each direction. Note that this abstraction does not mandate that an underlying bi-directional transport cannot be utilized. For H.323, a single RTP session may account for both logical channels (i.e. A to B and B to A) and the concept of a logical channel maps directly onto a *session ID* from RTP. Data channels (such as T.120 [22]) are typically treated as bi-directional logical channels.
- **Conference control**: to provide the endpoints with mutual awareness in n-way conferences, determine conference-wide suitable capability sets, establish the media flow model between all the endpoints (which are then initiated by means of the media channel control). Conference control also provides administrative conference functions such as chair control, floor control, and roster notification.

4.3. Real-time transport protocol

The Real-time Transport Protocol (RTP) [15] is a protocol developed by the IETF (Internet Engineering Task Force) to allow transmission of (continuous) real-time information streams across IP-based networks. The Real-time Transport Protocol consists of two parts. RTP defines the common RTP header format to be used with real-time data transmission; the Real-time Transport Control Protocol (RTCP) provides a mechanism for tracking and accounting information about the media stream itself and the quality of the underlying network—which is achieved by some low-bandwidth information exchange in the background between sender(s) and receiver(s). Both protocols are carried in UDP datagrams.

Traditional circuit-switching networks provide bit or byte pipes to carry real-time (isochronous) information streams (such as ISDN or PSTN and the related recommendations for video telephony, H.320 and H.324).Transmission delays of information units are constant, implicitly providing intra-stream timing; appropriate multiplex protocols on such pipes guarantee inter-stream timing as well (e.g. maintaining the timing relationship between the audio and the video stream from a participant to provide lip synchronization). For packet-based transports such as the Internet the situation is different, as are the requirements on a transport protocol for real-time information. Hence RTP provides the following functions:

- Media streams are not carried bit- or byte-wise; rather an information stream is fragmented into packets, which are then carried as payloads in RTP packets (which in turn are sent as UDP packets). Dedicated payload formats define per media encoding how the respective information

stream is to be split into packets. An RTP header field indicates which encoding format is carried in the payload of the RTP packet.

- UDP packets are carried unreliably across an IP network: they may be lost, duplicated, and reordered. The transit delay of UDP packets is variable while capture and playback of real-time information streams typically is continuous. A sequence number and a timestamp in the RTP header allow receivers to determine the appropriate playback point for each information unit (packet) received, and thus preserve intra-stream timing. Taking into account additional control information and feedback from RTCP messages, receivers can determine the current inter-arrival jitter and derive the correct playback delay therefrom. RTCP timestamps also allow correlation between different media streams to achieve inter-stream synchronization.
- As UDP and IP used underneath RTP already provide multiplexing on a per packet basis, no separate multiplexing function is needed at the RTP layer to distinguish different media streams. RTP headers provide a transport-address independent indication of the origin of each RTP packet.

4.4. Summary of H.323 protocol phases

The activity of the various protocols constituting H.323 as described in this section, can be summarized in a sequence of phases some of which are repeatedly entered. Fig. 4 depicts a conceptual phase model for the operation of H.323 systems and associates certain functions with each of these phases. In a simplified model, phases 0 and 1 involve the H.225.0 RAS protocol that also becomes active during shutdown and for each reconfiguration implying changes in resource utilization. Phase 2 comprises H.225.0 call signalling which may also be involved in phases 5 and 6. H.245 is active during phases 3 and 5 and is also used to terminate a call (phase 6). Media exchange based upon RTP and RTCP is carried out in phase 4.

The following section gives an overview of the protocol procedures followed for setting up calls and conferences in various modes of operation.

5. Operating scenarios

The H.323 protocol specification covers a wide range of operating scenarios: simple point-to-point calls are included as are multipoint conferences. The latter may be created either by ad-hoc expansion of a point-to-point call or by using MCUs to host conferences. Any number of the terminals in a call or conference may be located on non-IP-based networks (such as ISDN or PSTN) and be included in the H.323 call/conference via dedicated Gateways. In all of the aforementioned scenarios, Gatekeepers may be involved in address resolution and admission

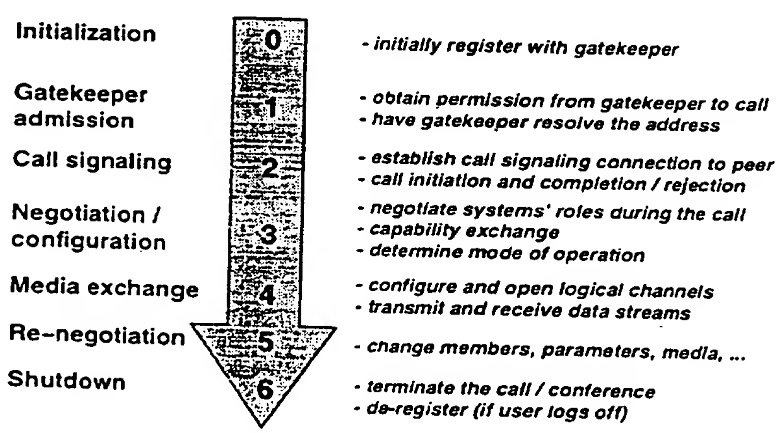


Fig. 4. H.323 Protocol phases.

control as well as in call signalling and conference control.

In all cases, the involved H.323 components follow the same overall protocol phases as depicted in Fig. 4 above. Phases 0, 1, as well as parts of phase 6 are only applicable if a Gatekeeper is present in the network configuration; phase 5 only applies to calls with dynamic encoding changes, invocation of supplementary services, and to multipoint conferences. In the following subsections, the H.323 protocol operation for simple point-to-point calls (phases 2, 3, 4, and part of phase 6) and for multipoint conferences (involving phases 2 through 5) are described as is the principal Gatekeeper operation (phases 0, 1, and 6).

Calls via Gateways to endpoints on other networks are a straightforward extension of point-to-point calls, with the Gateway acting as endpoint from the H.323 perspective. In such cases, the Gateway translates call signalling, conference control, media packetization, and encoding. The basic operation is the same as in simple H.323 calls, the mapping and procedural details are beyond the scope of this paper and hence are not discussed any further.

5.1. Point-to-point call establishment

A simple point-to-point call without a Gatekeeper shall serve as a starting point to illustrate the call procedures defined by H.323. Assume a scenario with two endpoints A and B, with A calling B. Then A initiates the call by first making a TCP connection to the *well known port* for H.323 (port 1720) at B's IP address; this connection is used to carry all the H.225.0 call signalling messages. A sends a SETUP message to B indicating the desire to place a call along with various call parameters. B typically first responds with an ALERTING message thereby indicating that the user is being notified ("the phone is ringing"), followed by a CONNECT message as soon as the user answers. As part of this exchange, A and B also send an ephemeral (dynamic) port number to be used for the H.245 connection—which may be established at any point in time during or after this exchange. After setting up the H.245 connection, virtually all the protocol activity takes place on the H.245 connection. There may be no further reason to use the Q.931 connection, which may be closed, but in practice is typically left up. Once the audio (and video) codecs and parameters have been negotiated, exchanging H.245 OpenLogicalChannel messages and the respective acknowledgments creates media streams. This sequence passes the transmitter's RTCP address and port number as well as the receiver's RTP and RTCP address and port number for a particular media stream (for example, audio or video). Recall that each channel is logically considered to be one way and, therefore, for two elements to exchange audio, two logical channels in opposite directions need to be opened. An H.323 call may be terminated by either endpoint sending an H.245 EndSessionCommand. An H.323 call is also terminated when the H.245 control connection is lost.

5.2. Multipoint conferencing with H.323

Teleconferences—pure audio as well as multimedia—are typically convened in either of two ways:
1. by ad hoc expansion of a point-to-point call to a multipoint conference by adding one or more participants; or
2. by means of pre-planned conference with the necessary resources set aside in advance to the start of the conference.

Both modes of operation are supported by H.323 using the same principal mechanisms for tightly coupled conferences [7]. H.323 uses the notion of a Multipoint Controller (MC) as the central entity that coordinates behavior of all the endpoints in a conference. The MC is elected during call establishment; once in place, the MC role does not change location for the duration of the conference. It may be located in any of the participating terminals (or Gateways), in a Gatekeeper, or in a special-purpose device for conferences such as an MCU.

For expanding a point-to-point call in an ad-hoc fashion into a multiparty conference, the entity hold-

[7] In order to additionally accommodate large-scale conferences, a model has been developed that allows co-existence of a tightly controlled core of H.323 participants with an arbitrarily large audience which is only loosely-coupled to the conference core. This enhancement is described in Section 6.2.

ing the MC places an outbound call to the participant(s) to be invited. This invitation may be triggered by any of the current participants by sending an appropriate call-signalling request to the MC. Incoming calls received by any of the terminals in a call or conference may be redirected to the MC so that the calling party can be included in the conference as well.

Pre-planned conferences are based on dedicated conferencing devices—e.g. MCUs or special Gatekeepers—to "host" the conference. Participants connect to such a dedicated device by either directly specifying its transport or alias address and then naming the conference they want to participate in. Alternatively, H.323 supports the notion of conference aliases that may be provided to the Gatekeeper, which then directs the call to the appropriate MCU. All functions of ad-hoc conference expansion to bring in additional participants are supported for pre-planned conferences as well, and are based upon the same mechanisms.

Independently of the manner by which a conference was initiated or where the MC is located, the data distribution in an H.323 conference may follow two different models:

- Centralized: the terminals send their audio/video/data streams to an MCU (MP) which then performs mixing and/or switching of the media streams and redistributes the resulting streams: individually to each terminal via unicast or commonly to all terminals via multicast.
- Distributed: each terminal transmits its media streams directly to all other terminals which are responsible for reception, decoding, and mixing/composition of these streams for local presentation; the media streams may be distributed via multicast to all peers or individually to each one via unicast (multi-unicast mode).

Within a single conference, these modes may arbitrarily be combined: different modes may be employed for different media, for different endpoints, etc.

5.3. Basic model for gatekeeper interaction

As indicated previously, endpoints are required to apply to Gatekeepers before claiming any resources in the network environment if they operate in a Gatekeeper-controlled environment. In order to determine if this is the case, endpoints attempt to register with their Gatekeeper. This registration is performed in two stages. Initially, the endpoint discovers a Gatekeeper that is willing to accept its registration either by querying a (set of) pre-configured Gatekeeper(s) with a GRQ message via unicast or multicasting the message to a well-known multicast address. Secondarily, the endpoint selects one of the Gatekeepers willing to accept a registration and registers its user aliases, transport addresses for call establishment and other parameters with an RRQ message. When shutting down, an endpoint de-registers from its Gatekeeper by means of a URQ message.

When an endpoint wants to place or answer a call, it queries the Gatekeeper by sending an ARQ message. The Gatekeeper accepts it by providing a transport address for establishment of the call signalling channel in the response (ACF); alternatively, the Gatekeeper may reject the ARQ by sending an ARJ thereby preventing the endpoint from proceeding with the call. When in a call, an endpoint may also have to contact the Gatekeeper to request changes in its resource utilization (via the BRQ message). Upon ending a call, an endpoint notifies its Gatekeeper by means of a DRQ message.

When an endpoint asks its Gatekeeper with an ARQ for permission to place or answer a call, the Gatekeeper may enforce one of two call models currently defined in H.323. The Gatekeeper may decide to allow the two endpoints to communicate directly with one another (*direct call model*). For the caller, this is done by returning the call signalling address of the called endpoint, for the callee, this is done by simply acknowledging the admission request. In this case, the call signaling connection and the H.245 connection run directly between the two endpoints. Alternatively, the Gatekeeper may keep local control over the call (*Gatekeeper-routed call model*) by having the call signaling connection as well as the H.245 connection routed through itself. On the calling side, this is achieved by returning the Gatekeeper's own call signaling address to the caller (rather than the remote endpoint's one). On the called side, the Gatekeeper explicitly requests a redirection of the call signaling connection thus requiring the caller to tear down the call signaling connec-

tion and re-establish it to the Gatekeeper of the called endpoint. The Gatekeeper-routed call model allows the Gatekeeper to keep track of the calls, act as an MC, and/or provide supplementary and other value-added services.

6. Recent enhancements

With over a year's worth of commercial development and deployment, IP Telephony has come to the forefront as one of the important applications for H.323 signaling. A result of this emergence has been a number of enhancements to H.323. The highlights of these enhancements include:

- Single roundtrip call connection sequence. In version 2 of H.323 the call establishment sequence is shortened by defining a procedure to simultaneously signal capabilities and propose the opening of logical channels in a single message to the callee. The callee then selects the media channels to receive and opens its own channel(s) to the caller in a single response. Hence a single call signaling message exchange suffices to start media streaming in both directions.
- H.245 Tunneling. H.323v2 allows H.245 messages to be carried within call signaling PDUs. This allows the TCP connections between entities to be reduced, in addition to allowing concurrent Q.931 and H.245 signaling.
- Extended addressing/alias types. H.323v2 enhances the variety of aliases that are allowed for call establishment. In particular, alias names for conferences and URLs are explicitly supported by the enhanced scheme (and may be explicitly distinguished without textual conventions on the alias' contents).
- Redundant/backup gatekeeper addressing. To provide seamless system operation even in the event of component failures, H.323v2 allows users to register with multiple Gatekeepers (primary and backup ones).
- "Follow-me" destination addressing. The version 2 Registration messages have been augmented to include a sequence of alternative transport addresses that might be utilized to contact the endpoint. A Gatekeeper may provide a list of alternate endpoints back or the Gatekeeper may mask this from the calling endpoint. In either case, the extra addresses can be polled to attempt call connections. By convention the order of preference is the ordered sequence.
- User level authentication/authorization. Utilizing new H.245 messagesthat were added to support the H.235 [2] framework (see next section), applications may exchange digital certificates. By issuing application explicit challenges and requesting specific certificate types, the protocol can support end-to-end authentication and related authorization. In practice this requires coordination with the local implementation to provide interactions with a human user (e.g. entering PIN numbers or approving of certificate contents).

These point enhancements along with newer peer protocols such as H.235 and H.332 portend to continued usefulness in new areas for H.323. The H.450.x series of recommendations [6] have been derived from the QSIG [8] standards and thus easily interface to existing PBX equipment. H.450.1 defines a framework for extending call control functions to provide higher level and more complex call services. The H.450.x series defines a remote procedure call scheme and initially describes a small set of functions such as call transfer and call forwarding. These functions may be provided by endpoints but also (similar to PBXs) in dedicated elements such as Gatekeepers. The H.450.x services and protocols are kept open to allow for easy future expansion by standardized as well as vendor-specific services.

6.1. H.235: the H.323 security framework

As with all communication applications, provision of security features is of crucial importance for H.323, particularly for global deployment. Designing security services for H.323 systems provides a number of challenges. Shared, packet networks require specialized media privacy to attain the perceived and expected protection offered by the circuit networks. Typical packet networks are lossy communication

[8] QSIG is an international standard which defines a signaling system in Private Integrated Service Networks (PISN). This is a generic term used to describe various types of voice networking equipment/services such as PBXs or CENTREXs.

environments offering additional challenges for security services. For example media encryption should not rely on a stream cipher across multiple RTP packets. Finally, limited resources such as Gateways or the media content itself must be protected from unauthorized use.

H.235 [2] is one of the newest ITU-T H.323 related recommendations, officially titled "Security and encryption for H series (H.323 and other H.245 based) multimedia terminals." This recommendation provides a general security framework that may be incorporated by many multimedia systems including H.323. H.235 "describes enhancements within the framework of the ITU H.3(XX) specification series, to incorporate security services such as *Authentication* and *Privacy* (data encryption). The proposed scheme is applicable to both simple point-to-point and multi-point conferences for any terminals which utilize H.245 as a control protocol." [2, p. iv] Recommendation H.235 describes a number of generic messages and procedures, which may be utilized to provide all the essential security services for interactive communications including authentication, privacy and integrity. The recommendations H.225.0 version 2 and H.245 version 3 include the necessary message extensions to enable the services described in recommendation H.235.

H.235 encompasses three phases of communication: call admission, call establishment and control, as well as conference control and media exchange (RAS, Q.931, and H.245/RTP, respectively). The framework described in H.235, reuses applicable protocols that exist such as Transport Layer Security (TLS) [8] or Internet Protocol Security (IPSEC)[9–14]. During each phase of an H.323 call, the H.235 security services applied to this phase may be separately negotiated—although the underlying cryptographic mechanisms are often related. As Fig. 5 shows, each sequential phase of an H.323 call (indicated by the "pipes") may be operated with a different set of security services enabled. In all cases, the type and level of authentication, integrity, and confidentiality may be negotiated (either within TLS, IPSEC, or explicitly in H.235).

The following subsections describe the security mechanisms available in the respective phases.

6.1.1. Call admission

RAS signaling between an endpoint and a Gatekeeper utilizes UDP and therefore TLS may not be used. In many instances, user authentication during registration (i.e. input for identification and challenges) make IPSEC usage impractical. RAS messages with H.235 extensions enable a number of

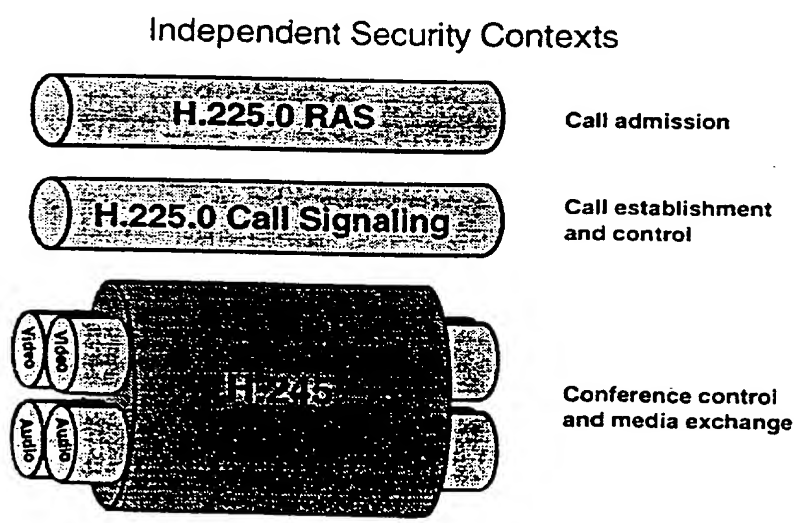


Fig. 5. Communication phases distinguished by H.235.

authentication methods between and endpoint and a Gatekeeper. ISO algorithms [18–21] provide the procedures for authentication assuming that there is a shared secret (e.g. password) or a common public-key certificate hierarchy between an endpoint and a Gatekeeper. For situations in which there is no shared secret, a Diffie/Hellman exchange may be used to establish key material for subsequent encryption or signatures. RAS messages may be generated with an integrity check value to provide tampering indications. There are no standard mechanisms to provide for RAS confidentiality (beyond those possibly supplied by the underlying transport).

6.1.2. Call establishment and control

Call establishment security services may be provided by the underlying transport session, in which case no explicit in band signaling is required. The well-known port 1300 may be used by H.323 entities to establish a Transport Layer Security (TLS) connection for call establishment and control (Q.931) signaling. The call establishment and control phase may be protected by TLS, IPSEC, or with digital certificate technology. These security mechanisms may provide authentication, confidentiality and integrity, thus specific H.235 signaling may not be needed. Authentication is either provided by the transport or through a cryptographic link (a signed security *token*) to the authentication which occurred during the call admission via H.225.0 RAS before. Q.931 messages do not have standard integrity check values. During this phase, H.235 security *tokens* may be utilized to provide *authorization*.

To provide a policy mechanism for *authorization* (which should be based on appropriate authentication) specific *tokens* are be passed with cryptographic links to their owners. For example, an IP telephony service operator might require a specific digital certificate signed by one of its Gatekeepers to be presented by a caller anytime a set of Gateways is utilized. All of the signaling and payloads required to accomplish this (and many more complicated scenarios) may be invoked within H.235/H.225.0 during the call initiation and establishment phases.

6.1.3. Conference control and media exchange

As with the call establishment, H.245 may utilize either TLS or IPSEC to provide security services. Independent of the operation of H.245, media encryption algorithms, modes and parameters are communicated by utilizing well-defined identifiers in the form of Object Identifier tags. This allows for easy implementation of future enhancements to the architecture. The identification mechanism also allows the full array of publicly known algorithms along with any proprietary methods to be signaled in a standardized, recognizable manner.

Encryption of media is used within the RTP streams to provide reasonable performance and flexibility in multipoint situations. The session keys that are used to encrypt the media may be distributed in a number of ways by utilizing H.245 signaling. For example, the session key itself may be protected with the transient shared secret that the elements established at the beginning of communications or may be conveyed to the peer(s) by using public key cryptography. H.235 allows refreshing the session key on the fly, thereby enabling "breaches" in security or expulsion from a multipoint conference to be accomplished.

Facilities for a challenge/response exchange between users and the network and end to end-users are provided. Within H.323, these facilities are enabled by H.245 PDU exchanges between peers.

6.1.4. Operational aspects

Unlike other aspects of communications, such as call control and transport protocols, security technology is significantly influenced by non-technical factors. One of these environmental factors that influenced the development of H.235 will continue to impact its deployment: politics. Due to the nature of the subject, political issues along international and other boundaries, are prominent factors: countries limit distribution of (certain types of) security technology, ban or constrain its deployment within a country, etc. The largest manifestation of these issues within H.235 is the requirement to negotiate all of aspects of security: for example there are no requirements for a base level cryptographic algorithm to be supported. This resulted from the lack of international consensus concerning which algorithms to employ. Instead of performing the work in the ITU-T, it is expected that market segments and/or vertical applications will develop fixed "profiles" for complete cryptographic interoperability.

6.2. H.332: loosely-coupled conferencing with H.323

The H.332 recommendation [5] extends the tightly controlled model of H.323. Where H.323 encounters practical limits due to its tightly coupled model, H.332 provides an architecture and the necessary protocols for very large-scale operations. The basic conference model that H.332 assumes, is that of a panel-style conference: a single presenter or a small group of participants (the panel) provide the multimedia contents that is distributed to a virtually arbitrarily large audience. As depicted in Fig. 6, the core panel consists of a H.323 conference and is "surrounded" by a large number of RTP receiving terminals. These RTP receiving terminals may be H.332 terminals or other RTP/RTCP capable terminals that have external means to understand how to connect to the conference.

Establishment of the panel and interactions among its members are tightly controlled using the conventional control mechanisms of H.323. Administrative control of the conference is provided through "social protocols" or through H.323 chair- and floor control mechanisms. H.323 chair-control gives special privileges to the conference chairperson and if chair control is active, any panel member who wants to talk (or send video) must first request the floor from the chairperson. Outside the panel, the participants are passive; they are essentially receivers who are, by default, not allowed to interact. If they wish to interact they request join the panel or wait to be invited by someone on the panel, just as would occur in a conventional H.323 conference. Admission to the panel may be determined by some conference policy implemented in the MC and/or may be decided upon by the chairperson on an individual basis. The chairperson may also force members to leave the panel in order to make room for new ones.

While the H.323 protocols are re-used to establish the panel and change its members, these mechanisms for establishing connections and negotiating operating modes at the start of a conference are cumbersome and impractical for conferences involving an arbitrarily large number of participants. In such cases,

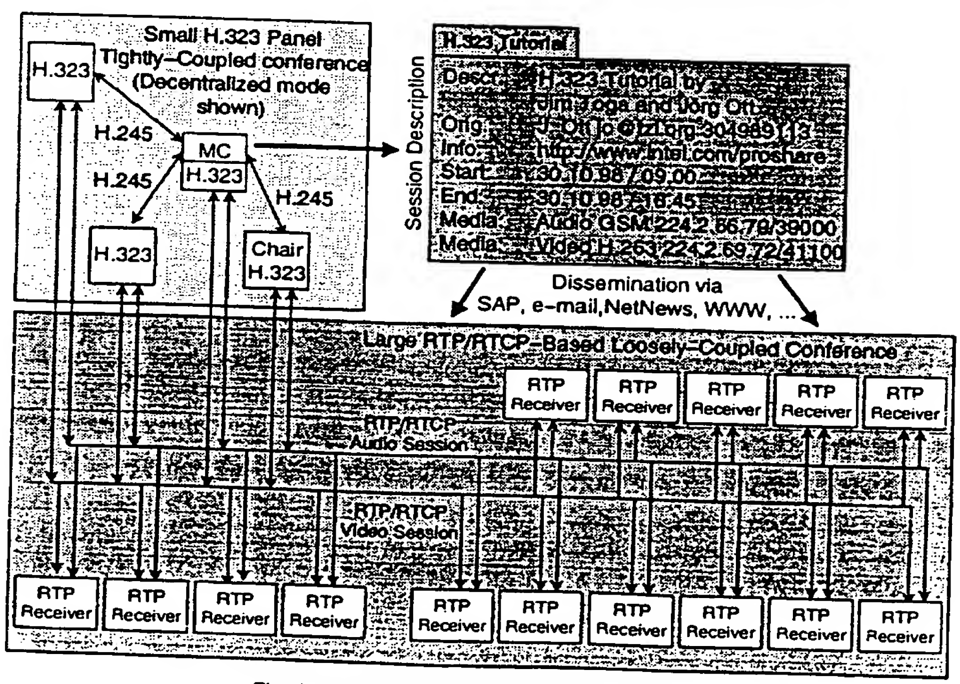


Fig. 6. Model of an H.332 panel-style conference.

the information required to setup a large conference must be disseminated well before the start of the conference. Large conferences are usually planned and pre-announced—examples include presentation to a large geographically dispersed audience, distance learning, etc. If a conference is pre-announced, then the conference modes of operation (such as multicast addresses, media capabilities) may also be pre-announced to all potential participants thereby eliminating the need for negotiation at conference startup.

For the announcing of H.332 conferences and their associated parameters, recommendation H.332 utilizes the format described the Session Description Protocol (SDP) [17] developed by the IETF to describe conference information. The Session Announcement Protocol (SAP) [16], web pages, Netnews groups, and even email, may be used to convey such conference descriptions; the specific manner of disseminating this information is outside the scope of H.332. The SDP format is enhanced by a few H.323-specific attributes including addressing information that allows members of the audience to contact the MC if they want to join the panel.

The media exchange/dissemination in an H.332 conference is accomplished via RTP/RTCP as transport for audio and video information. The panel may operate in any H.323 mode: centralized, decentralized, or hybrid. Outside the panel, however, multicast is used for information dissemination in order to provide the scalability required for the H.332 conference. In addition to H.323 conference control mechanisms that provide mutual awareness among the panel members, RTCP reports are evaluated to obtain a rough understanding of the conference size and the "identities" of its (non-panel) members.

As with H.323, the support for audio is mandatory in H.332, while video and data are optional. If any of the optional media is supported, the ability to use a specified common mode of operation is required so that all terminals supporting that media type can interoperate. H.332 allows more than one channel of each type to be in use in the same manner as H.323 does.

For pure audio-visual conferences, the design choices of H.323 and H.332—i.e. re-use of existing protocols, SDP, SAP, and RTP/RTCP—allow seamless interoperability even with non-H.332-capable endpoints, the most prominent examples being the variety of Mbone conferencing tools available today (such as *vic* [23] and *rat* [24]).

6.3. Future work

While the H.323 series of Recommendations provides a sound technical foundation for multimedia communication in IP networks including IP telephony as special case, a variety of (global) infrastructure aspects need to be dealt with accompanying the further development of the technical core protocols. The responsible ITU-T working group as well as the ETSI TIPHON project have taken up complementary work items towards a further completion of the work. As even an outline of the individual efforts are beyond the scope of this paper, the section is restricted to very briefly listing the work items currently under development:

On the ITU-T side, current standardization efforts include further completion of the supplementary services provided by H.323; improved support for trunking (i.e. the use of H.323 in telephony backbones); inter-gatekeeper protocols (for communication within as well as across administrative domains); support for remote device control; seamless inclusion of facsimile transmission utilizing H.323 control; and provision of appropriate Management Information Bases (MIBs) for H.323 systems and protocols.

Within ETSI, on-going efforts include the development of a suitable numbering plan for IP telephony; security profiles for both consumers and service providers. Infrastructure services including billing, and accounting mechanisms for a variety of call scenarios are further efforts as are work items such as coordination of clearinghouse services to Quality of Service measurements.

7. Conclusion

This paper has provided an overview of H.323 and its associated recommendations by presenting system components, protocols, and modes of operation as well as pointing out recent development directions. The H.323 system provides a powerful and flexible system for tightly controlled, interactive, real-time, multimedia communications. The factors

that allow the protocols to easily bridge data and voice networks also make H.323 scalable. For example, the dynamic exchange of capabilities allows communication modes to change during a call if needed and adapt to any (changes in) environmental or endpoint constraints. Distribution of media processing across different Gateways or MPs contribute to scalability and bandwidth or processing flexibility. The elements that make up an H.323 network (terminals, Gateways, Proxies, Gatekeepers, and MCUs) enable the deployment of H.323 in a variety of physical topologies and operational models.

Since its early development stages, the H.323 series of recommendations has gained broad industry attention and support. The ongoing product development in the industry on a very broad basis—including a wide range of communication systems from simple point-to-point telephony to rich multimedia conference systems—demonstrates this endorsement. The scale and success of frequent interoperability test events—sponsored by the International Multimedia Teleconferencing Consortium (IMTC)—emphasize the viability of H.323 as a cross-vendor platform for interactive real-time communications in IP-based networks. Through permanent effort by the ITU-T study group responsible for H.323, the recommendation continues to be evolved and adapted to address new technical issues, match new situations, and meet new customer needs. Particularly with last year's unparalleled efforts to efficiently accommodate IP telephony applications—the *killer application* per se—and with the current work focus on a globally scalable infrastructure, H.323 is well-advanced on its way towards enabling ubiquitous, interpersonal multimedia communications in an integrated global network.

References

[1] H.225.0, Call Signaling Protocols and Media Stream Packetization for Packet Based Multimedia Communications Systems, ITU-T Recommendation, 1998.

[2] H.235, Security and Encryption for H Series (H.323 and other H.2456 based) multimedia terminals, ITU-T Recommendation, 1998.

[3] H.245, Control Protocol for Multimedia Communication, ITU-T Recommendation, 1998.

[4] H.323, Packet Based Multimedia Communications Systems, ITU-T Recommendation, 1998.

[5] H.332, H.323 Extended for Loosely-coupled Conferences, ITU-T Recommendation, 1998.

[6] H.450.1, Generic Functional Protocol for the Support of Supplementary Services in H.323, ITU-T Recommendation, 1998.

[7] Q.931, Digital Subscriber Signaling System No. 1 (DSS 1) – ISDN User-Network Interface Layer 3 Specification for Basic Call Control, ITU-T Recommendation, 1993.

[8] T. Dieks, C. Allen, The TLS Protocol Version 1.0, draft-ietf-tls-protocol-03.txt, Work in Progress, Internet Engineering Task Force, 1997.

[9] D. Harkins, D. Carrel, The Resolution of ISAKMP with Oakley, draft-ietf-ipsec-isakmp-oakley-04.txt, Work in Progress, Internet Engineering Task Force, 1997.

[10] R. Atkinson, Security Architecture for the Internet, RFC 1825, Internet Engineering Task Force, 1995.

[11] R. Atkinson, IP Encapsulating Security Payload (ESP), RFC 1827, Internet Engineering Task Force, 1995.

[12] R. Atkinson, IP Authentication Header, RFC 1826, Internet Engineering Task Force, 1995.

[13] D. Maughan, M. Schertler, M. Schneider, J. Turner, Internet Security Association and Key Management Protocol (ISAKMP), draft-ietf-ipsec-isakmp-08.text, Work in Progress, Internet Engineering Task Force, 1997.

[14] H.K. Orman, The Oakley Key Determination Protocol, draft-ietf-ipsec-oakley-02.txt, Work in Progress, Internet Engineering Task Force, 1997.

[15] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, RTP: A Transport Protocol for Real-Time Applications, RFC 1889, Internet Engineering Task Force, 1996.

[16] M. Handley, SAP: Session Announcement Protocol, draft-ietf-mmusic-sap-00.txt, Work in Progress, Internet Engineering Task Force, 1996.

[17] M. Handley, V. Jacobson, SDP: Session Description Protocol, RFC 2327, Internet Engineering Task Force, 1998.

[18] ISO/IEC 9798-2:1994, Information Technology – Security Techniques – Entity Authentication – Mechanisms Using Symmetric Encipherment Algorithms.

[19] ISO/IEC 9798-3:1993, Information Technology – Security Techniques – Entity Authentication Using Public Key Algorithm.

[20] ISO/IEC 9798-4:1995, Information Technology – Security Techniques – Entity Authentication – Mechanisms Using A Cryptographic Check Function.

[21] ISO/IEC 11582, Information Technology – Telecommunications and Information Exchange Between Systems – Private Integrated Services Network – Generic Functional Protocol for the Support of Supplementary Services – Inter-exchange signaling procedures and protocol.

[22] T.120, Data Protocols for Multimedia Conferencing, ITU-T Recommendation, 1996.

[23] S. McCanne, V. Jacobson, Vic: a flexible framework for packet video, Proc. ACM Multimedia '95, Berkeley, CA, 1995.

[24] V. Hardman, A. Sasse, I. Kouvelas, Successful multi-party audio communication over the Internet, Comm. ACM 41 (5) (1998) 74–80.

James Toga received a B.Sc in Chemistry from Tufts University (1983) and a M.Sc in Computer Science from Northeastern University (1992) in the United States. Before joining Intel, he was the principal engineer on StreetTalk[9] Directory Service with Banyan Systems where he designed and developed a generation of the Yellow Pages service. Presently, he is a senior software architect for the Standards and Architecture Group in the Intel Architecture Labs. He coordinates product groups giving guidance on architecture and standards. His primary tasks are H.323/Internet Telephony, Directory, and real-time security issues. Outside of Intel, Mr. Toga develops standards and standards-based products within ITU-T, ETSI, IETF, and IMTC. He is Editor of the ITU-T documents, "H.323 Implementers Guide" and "Recommendation H.235". Mr. Toga also chairs the IMTC "Packet Networking Activity Group" and the H.323 Interoperability Group. Mailto: jim.toga@intel.com.

[9] All other trademarks are the property of their respective owners.

Jörg Ott received his diploma in Computer Science in 1991 and his Doctor in Engineering (Dr.-Ing.) in 1997 from Technische Universität Berlin. He also holds a diploma in Economics from the Technische Fachhochschule Berlin (received in 1995). His interests are in protocol and system architectures for multipoint communications and multimedia conferencing, including Internet Telephony as special interest area. Dr. Ott has been affiliated with the Berlin-based TELES AG since 1989 where he was system engineer, later on project manager, and finally became an external technical advisor. From 1992 to 1997 he held a research position at Technische Universität Berlin working on interactive multipoint and multimedia communications. Since 1997, he is "Wissenschaftlicher Assistent" in the Research Group for Computer Networks at the Universität Bremen. In the ITU-T, he is Associate Rapporteur for coordination between the ITU-T and the IETF with respect to Multimedia conferencing and Internet Telephony and is also editor of two new Annexes to Recommendation H.323 addressing special-purpose terminals. Since 1997 he is co-chair of the MMUSIC working group of the IETF. Mailto: jo@tzi.uni-bremen.de.